General Station Troubleshooting

Ed Himwich, Dan Smythe,
& Brian Corey

Overview

- This course covers station VLBI equipment tests that are more extensive than those described in the *Experiment Pre-checks and Operations* workshop.
- These tests should be carried out
  - after a major equipment change,
  - prior to a major, multi-day observing campaign such as CONT11, or
  - at least annually, and preferably more frequently.
- For VLBI systems with a Mark IV decoder, data acquisition and analysis for most tests are automated through the *systests* FS procedures and perl scripts, which are described and demonstrated in this workshop.

Antenna pointing offsets

- Measure pointing offsets over the full az/el range of the antenna, by observing multiple sources over a wide range of declinations over a 24-hour period
  - Source size should be < X-band beamwidth
- Verify that pointing errors are
  - typically < 0.1 FWHM X-band beamwidth
  - always < 0.3 FWHM X-band beamwidth

System Equivalent Flux Density

- Measure SEFD in each frequency channel on a strong source of known flux density with angular size < X-band beamwidth.
  - Use FS **onoff** command.
  - Or measure system temperatures on and off source separately, and calculate:

    SEFD = (source flux density) x Tsys_off

    / (Tsys_on – Tsys_off)
- Repeat measurements, on same source and other sources, to check consistency.

Phase cal power level

- Measure fractional power contributed by phase cal to total system power in each channel in two ways:
  - Measure broadband phase cal power. Options:
    - Use *systests* **pcalpowera** or **pcalpowerb** procedure, followed by **pcalpower** script.
    - Using **tpical** and **tpi** commands, measure baseband total power levels with phase cal on and off, respectively; then compute fractional power = (tpical – tpi) / (tpi – tpzero) .
  - Measure amplitude of pcal tones in each channel with
    - *systests* **pcalamp** procedure & script,
    - Mk4 decoder **pcal** command, or
    - FS **dqa** command for VLBA DAR (works only at 10 kHz).

Phase cal power level – cont'd

- Check for consistency between results of two methods.
  - *systests* script '**pcalpower –b**' option converts fractional pcal power estimates into equivalent pcal amp values, for comparison with results of *systests* **pcalamp** results.
  - If *systests* is not available, convert from fractional pcal power to amp via
    $$\text{amp} = 1000 \sqrt{(\text{pcal power}) / \text{BW(MHz)}}$$
    (Factor of 1000 occurs because amp is reported in units of 0.1%.)
- Pcal amplitudes significantly smaller than predicted from pcal power may indicate loss of power into pcal tone sidebands due to modulation of phase cal or LO.

Spurious phase cal signals

- Use *systests* **pcalspur** procedure & script, or Mk4 decoder **pcal** command, to measure amplitude of every pcal tone (or **dqa** to measure 10-kHz amp) in each channel with
  - phase cal on
  - phase cal off
    - by disconnecting 5 MHz signal to phase cal antenna unit
    - separately, by turning off ground unit switch, if available
  - phase cal on but receiver LO unlocked
- To dig deeper into noise, use low-frequency FFT analyzer to measure power of 10-kHz pcal tone in each channel, under same conditions as above.
  - If pcal tone used at correlator is not at 10 kHz, BBC/VC LO frequency must be shifted to put tone at 10 kHz.
- Ideally, power of each tone will drop by 50 dB when pcal is turned off, or LO is unlocked, compared with pcal on.

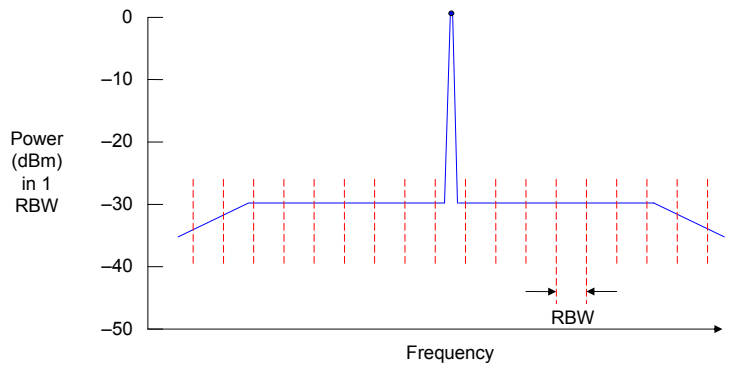
VC/BBC tests: image rejection & LO lock

- USB/LSB image rejection
  - Use *systests* **pcalimage** procedure & script (or Mk4 decoder **pcal** command or FS VLBA **dqa** command) or spectrum analyzer to measure phase cal amplitude in both sidebands of each channel.
  - Amplitudes in two sidebands at the same baseband frequency should differ by >20 dB (factor of >10 in voltage).
  - Measure at several frequencies spread across baseband, if possible.
- Check that LO locks over full frequency range:
  - 100-500 MHz for Mk4
  - 500-1000 MHz for VLBA

VC/BBC LO phase noise

- Measure LO phase noise at 2-4 widely separated LO frequencies
  - Test tone method –
    - Turn off phase cal.
    - Set VC/BBC LO to xxx.xx MHz.
    - Using a signal generator known to have low phase noise, inject a test tone at frequency xxx.xx + 0.01 MHz into IF.
    - Observe 10-kHz beat signal between test tone and LO in USB output on oscilloscope. Measure RMS jitter in zero-crossing time of signal, with scope triggered from an external, maser-synced time base.
  - Spectrum method –
    - Use method described in spectrum analyzer workshop to measure phase noise and RMS phase jitter.
- Compare measured jitter against VC/BBC LO specification:
  - Mk4: rms < 4° below 450 MHz, rms < 9° above 450 MHz
  - VLBA: rms < 2°
- Compare spectrum against reference spectrum.

Measuring LO phase noise on spectrum analyzer



• Phase noise = total power in two modulation sidebands, on either side of carrier

• RMS phase jitter (radians) = sqrt[(power in 2 sidebands) / (power in carrier)]

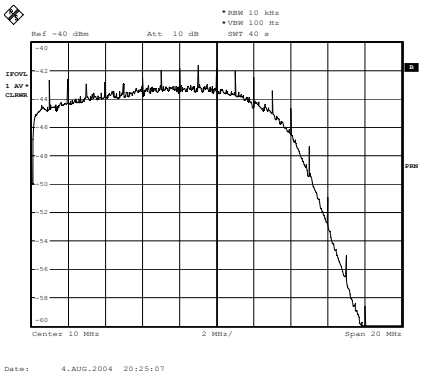
VC/BBC baseband bandpass shape

- Using spectrum analzyer, measure shape of baseband spectrum in each frequency channel, and compare against ideal.
- Ideal bandpass shape of baseband signal, whether measured at VC/BBC analog output or after digitization in sampler, is…
  - Flat to <1 dB from near DC to 0.8 x filter BW
  - Down by 3 dB at 0.9 x filter BW
  - Down by 7-8 dB at bandedge
- Shape may be affected by RF/IF filter characteristics, RFI, ripple due to impedance mismatches, etc.

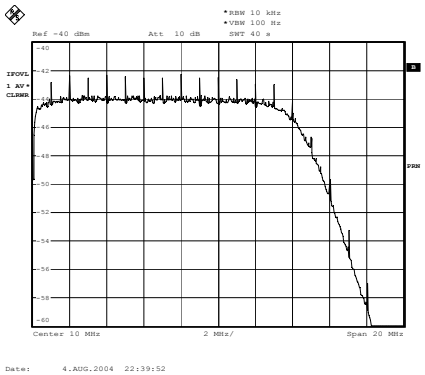
VC bandpass spectrum example: Wettzell

16-MHz-BW spectra measured with spectrum analzyer – 0-20 MHz span, 2 dB/vertical division

VC01 LSB – not so good!

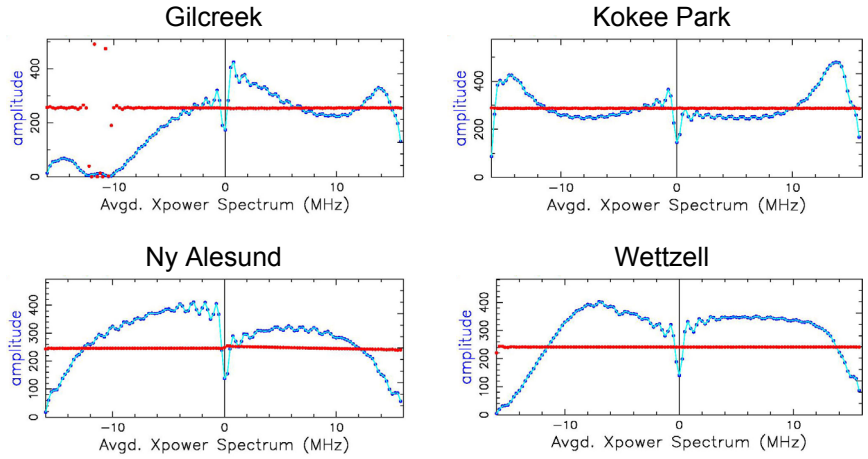


VC01 USB – good



Examples of spectra as seen at correlator

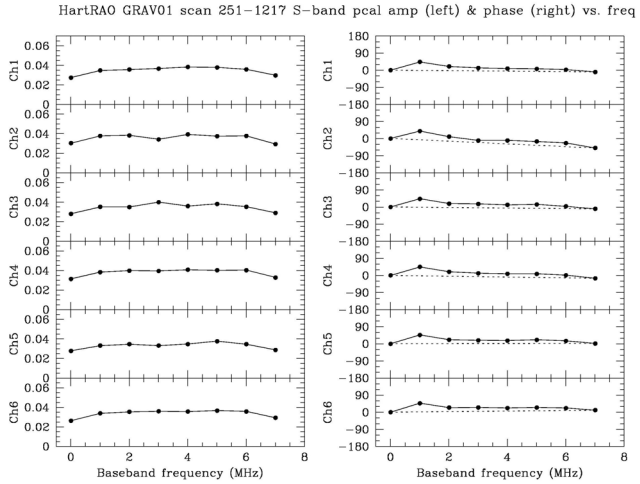
16-MHz-BW spectra for channel X1 LSB & USB in RD0504



Baseband spectrum from phase cal

- A low-resolution spectrum of each digitized baseband signal can be obtained from frequency dependence of phase cal amplitudes.
- Use *systests* **pcalamp** procedure & script, or Mk4 decoder **pcal** command, to measure amplitude of all phase cal tones in each channel.
- Phase cal amplitude spectrum is affected by same factors as continuum spectrum, as well as by
  - spurious signals
  - short-term phase coherence
  - image rejection, etc.
- Phase cal phase spectrum, which can be measured with Mk4 decoder **pcal** command, has additional information on spurious signals.

Phase cal amp & phase example: HartRAO



Phase shown after subtracting model (phase_10kHz + delay * freq * 360), with delay = −0.19 us.
Dashed straight line connects first and last phase points.

Sample state counts

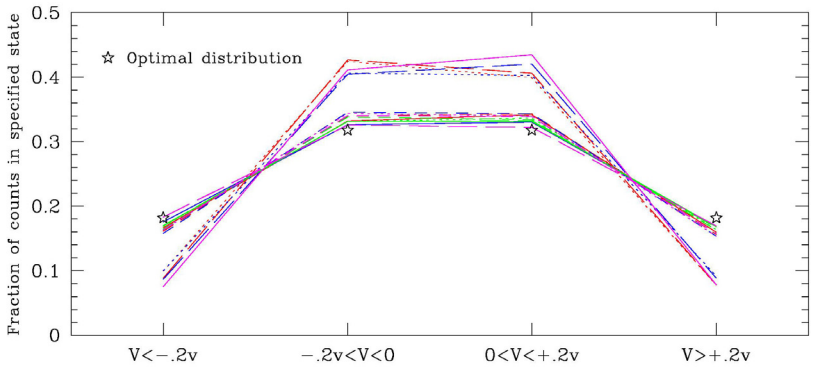
- Sample state statistics should be monitored via *systests* **samplestat** procedure & script, or Mk4 decoder **samples** command, to check for potential problems in analog or digital domain.
- Ideal distribution is

Sample state	*% of total samples*
– –	18%
–	32%
+	32%
+ +	18%

- Deviations from ideal may indicate…
  - Impedance mismatch in VC/BBC→formatter cable
  - No input to sampler
  - VC/BBC AGC circuit not working properly

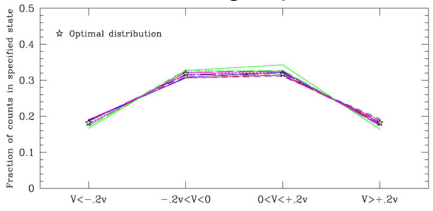
Example of state counts

Gilcreek RD0504 state counts for all 16 sidebands

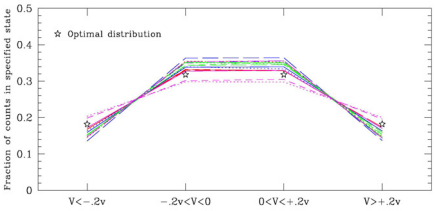


More examples of RD0504 state counts

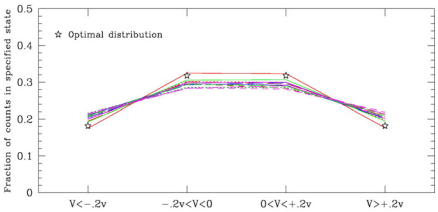
Algonquin



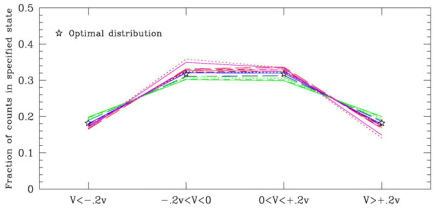
Kokee Park



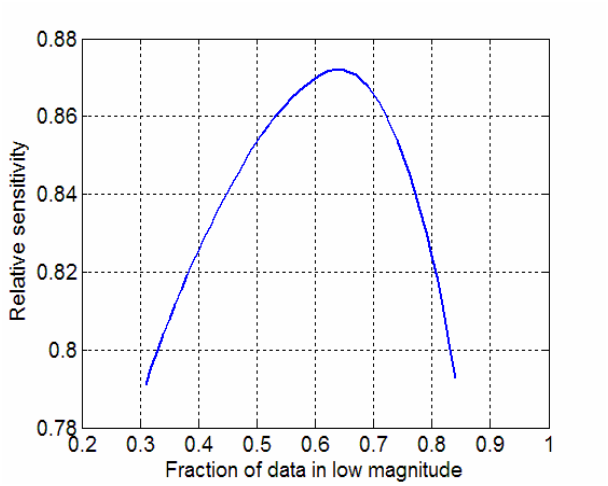
Westford



Wettzell



What happens if state count distribution deviates from the ideal?



Diurnal behavior

- Park antenna at zenith.
- Use *systests* **overnite** procedure, or comparable one of your own devising, to log at least the following quantities at 5-minute (or shorter) intervals for 24 hours:
  - Phase cal phases and amps in 2 X-band channels
  - Phase cal phases and amps in 2 S-band channels
  - Cable cal
  - Formatter-GPS time interval
  - Weather data
  - Receiver monitor points
  - System temperatures in all channels
- Run *systests* **plotlog** script on log to look for unexpected variations.

Short-term variations

- Park antenna at zenith.
- Use *systests* **rapid** procedure, or comparable one of your own devising, to log at least the following quantities at 15-second (or shorter) intervals for at least 10 minutes:
  - Phase cal phases and amps in 2 X-band channels
  - Phase cal phases and amps in 2 S-band channels
  - Cable cal
- Run *systests* **plotlog** script on log to look for systematic variations or unusual short-term variations.

Antenna orientation-dependent effects

- As the antenna is slewed slowly in azimuth and elevation over its full range of motion…
  - Use the FS to record cable cal and phase cal amplitude and phase rapidly, with at least 20 sets of data points per azimuth or elevation slew.
    - FS *systests* **rapid** procedure may be used to acquire data, and *systests* **plotlog** script to examine the logged data for unstable behavior.
  - Observe each IF with a spectrum analyzer and look for in-band RFI and strong out-of-band RFI.
  - Watch cable cal counter for jumps or rapid drifts.
  - Watch S and X phase cal phases on oscilloscope for jumps, rapid drifts, or unusual jitter that might be missed in FS log readings.

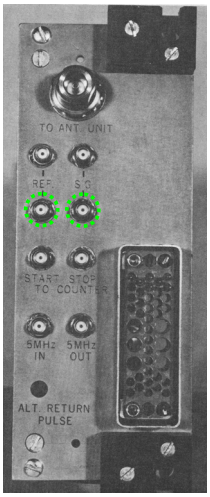
Cable cal

- Is cable reading stable to < 2 microsec peak-peak on time scales of 1-30 seconds?
  - It should not be necessary to operate counter in averaging mode to achieve this level of stability. Counter should be operated in single-sample mode.
- With test cable of known length inserted, does counter reading change by correct amount?
- On Mk4 ground unit, is phase meter between ¼ and ¾ of full scale?

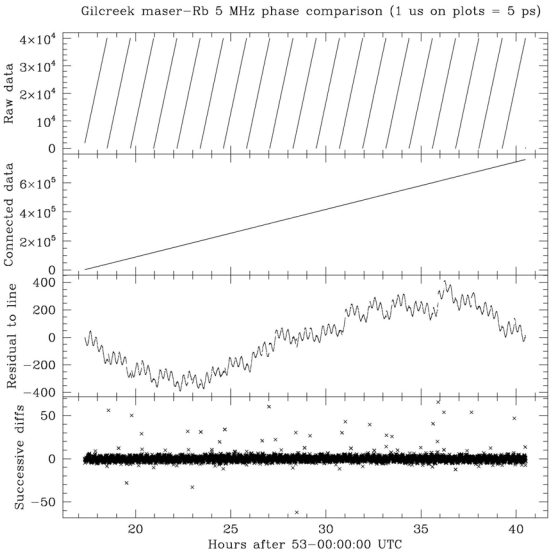
Phase comparators for reference frequency tests

- Phase instabilities in an LO or phase cal often result from instabilities at the reference frequency (e.g., 5 or 500 MHz).
- Vector voltmeters can be used to check the phase stability of a suspect signal against a known-to-be-good signal.
- At sites with a Mk4 delay cal ground unit, the relative phases between two 5 MHz signals can be measured by
  - Feeding them into the lower REF and SIG jacks on the rear panel (circled connectors in figure to right), and
  - Recording the cable cal counter readings in the usual way in a FS log.

Mk4 ground unit rear panel



Example of using delay cal ground unit for 5 MHz phase comparisons: Gilcreek maser-vs.-rubidium frequency standards



Mark 5A/5B checks

- Basic information on checking Mark 5A and 5B recorder performance is available in the notes for the *Experiment Pre-checks and Operations* workshop.
- Automated testing of a Mark 5A recorder in various modes, including 1 Gbit/second, can be carried out using *systests* tools:
  - Execute *systests* schedule **chk1024.snp** or **vchk1024.snp**.
  - Run *systests* script **chk1024** on the resulting log file to analyze the results.
- Additional information on testing a Mark 5B recorder is available in the notes for the *Mark 5 Advanced Topics* workshop.

Rack power

- Use oscilloscope to measure ripple and noise on DAT/DAR power supplies.
  - Low-frequency ripple should be < 10 mV pk-pk.
  - Total noise should be <100 mV pk-pk.
- Use multimeter to measure DC supply voltages *at the modules*, not at the power supply terminals, as there may be significant IR voltage drop from the supply to the module. Measured voltage should be within 0.1 V of design voltage.

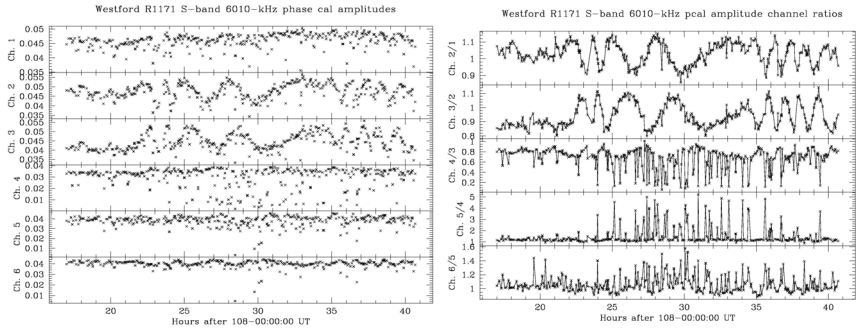
Meteorological sensors

- Temporal stability
  - Barometer should be stable to 0.1 mb over a few minutes.
  - Temperature should be stable to 0.1 °C over a few minutes.
- Barometer should be calibrated annually.
- Measure barometer offset relative to a local authority, e.g., airport.

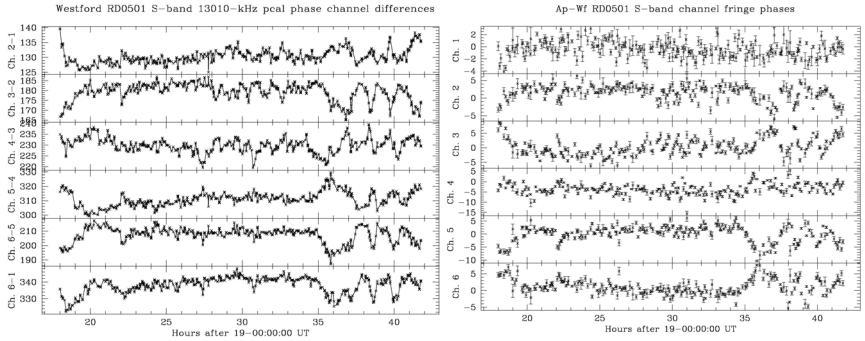
Complementary VLBI data checks at correlator

- Do fringe phases line up between channels?
- Are fringe amplitudes the same in all channels?
- Do the following quantities vary during scans, or between scans?
  - Fringe phase difference between channels
  - Fringe amplitude ratio between channels
  - Phase cal phase difference between channels
  - Phase cal amplitude ratio between channels
- Is the scatter in the estimated delays reasonable?
- Is there evidence of RFI?
- Etc.

Example: Westford R1171 S-band phase cal amplitudes extracted at correlator



Example: Westford RD0501 S-band phase cal and fringe phases from correlator



General Station Troubleshooting: "systests"

Brian Corey, Dan Smythe, and Ed Himwich

May 2011 TOW

Introduction

This write-up covers the "systests" suite of SNAP schedules and procedures libraries for acquisition and Perl scripts to analyze the results for many of the tests covered in the rest of this course. Most of these tests require that you have a Mark IV Formatter and Decoder with either a VLBA4 or Mark IV rack (VLBA5 and Mark 5 racks are also supported). The schedules and procedures require some basic customization at each station. Once customized, they largely automate the acquisition of the test data. The scripts automate the analysis producing plots.

The tests supported are: (1) band pass shapes, (2) baseband/video converter image rejection, (3) phase-cal power measurements, (4) phase-cal spurious signals, (5) sampler statistics, (6) "overnight" stability, (7) short-term stability, (8) orientation dependent effects, and (9) Mark 5A tests. A few of these tests do not require that you have a Mark IV Formatter and Decoder with either a VLBA4 or Mark IV rack (or VLBA5 or Mark 5 racks). Systems without these racks can still use test (3) which only requires a working "caltsys" procedure, and tests (6), (7), and (8) which can be modified to do a more limited test for other configurations. The Mark 5A tests will not work Mark 5B recorders, consequently they are not useful systems with VLBA5 or Mark 5 racks. Tests for Mark 5B recorders are covered in notes for TOW 2011 Operations Workshop: *Mark 5 Advanced Topics*.

The tests can be divided into functional groups: automated instrumental tests, interactive instrumental tests, and stability tests. The automated instrumental tests are: band pass shapes, baseband/video converter image rejection, sampler statistics, and Mark 5A tests. The interactive instrumental tests are: phase-cal power measurements and phase-cal spurs. It is possible to automate these tests if the station has computer control of necessary signals. The stability tests are: "overnight" stability, short-term stability, and orientation dependent effects.

An overview of installation, data acquisition, and data analysis are presented here. Two scripts that warrant additional discussion, "`chk1024`" and "`plotlog`" are covered in small additional section. After these sections, a copy of the "`systests.txt`" documentation file (found in `/usr2/fs/misc`) is appended followed by some examples of output.

Installation

The installation is covered in detail in the attached "systests.txt" file. The "systests" suite may have already been installed at some stations as part of a previous FS upgrade. The three basic steps are:

1. Copy the relevant files from /usr2/fs/systests to /usr2/proc and /usr2/sched.

2. Modify the station specific parts of /usr2/proc/systests.prc. You should use pfmed to make these changes. There are three types of procedure modifications.

> A. A few procedures just need to have the right set of comments uncommented depending on the type of rack being used: Mark IV or VLBA4. These procedures are: "tpisamp", "bbcmanx", and "bbcagx".
>
> B. A few procedures need to be modified to use local commands, if they exist, to turn on or off certain signals: phase-cal, LO 5 MHz, and phase-cal 5 MHz. If there is no local control for these signals, the default contents, which will prompt the operator to manually change the state of the signals, can be used. These procedures are: "pcalonsys", "pcaloffsys", "lounlocksys", "pcal5moffsys" and "pcal5monsys".
>
> C. Two procedures: "stsample" and "onsample" may require more specialized edits depending one what commands and equipment are available for short-term and overnight monitoring. Most of these changes should be straightforward; by default they try to monitor all the appropriate standard data items. If you don't have some of these items or the commands at your station are different, you will need to comment them out or change them. If you have other station specific items you would like monitored, you can add them as well.

3. For most of the analysis scripts you will need to have the Pgperl package installed. Instructions for installing this on a FS Linux 5, 6, & 7 distributions are included in "systests.txt". FS Linux 8 includes Pgperl.

Data Acquisition

The steps for data acquisition are streamlined. Only a summary is presented here. More details are available in "systests.txt". Except for the Mark 5 tests (covered at the end of this section), the general steps are:

1. Setup the equipment in the mode you want to do the test in. This is usually done by opening the procedure library that contains the setup procedure you want to use and executing. For example, this might look like:

```
proc=rd0505gc
setup8f
```

2. Open a log to record the test:

```
log=…
```

It is recommended that each test, and each time it is run, is recorded in a new log. You can optionally open the log before doing the setup so that the equipment configuration is recorded as well.

3. Open the “systests” procedure library and execute the test you want, for example:

```
proc=systests
pcalamp
```

4. Further tests in the same equipment configuration can be executed by repeating steps 2 and 3.

Mark 5 tests are organized as a schedule rather than a procedure, so each time they are started, they will open a log of the same name, chk1024.log or vchk1024.log, for example:

```
schedule=chk1024,#1
```

When the schedule is finished you should close and rename the log so that: (1) another test does not get added to the log and (2) the log has a name more mnemonic and easier to identify than the generic name of the schedule. For example for a test run at Svetloe on the 100th day of 2007, you might use: chk1024sv.07.100.log. Other ways to name the file to identify the contents uniquely and clearly can be used of course. For example, the commands could be:

```
log=station                                          (at a FS prompt)
mv /usr2/log/chk1024.log chk1024sv.07.100.log        (as a shell prompt)
```

Data Analysis

The data analysis is streamlined by having a single script to analyze the data from each test. For example, if you have the results of a phase-cal amplitude test in a log, renamed to, “pcalampsv.07.100.log” you can generate the plots for it using:

```
cd /usr2/log
pcalamp pcalmpsv.07.100.log
```

In general the names of the analysis scripts mirror those of the SNAP procedures and schedules that collect the test data; Table 1. gives the correspondence.

Test	SNAP	Analysis Script
Band pass shapes	pcalamp	pcalamp
BBC/VC image rejection	pcalimage	pcalimage
Phase-cal power	pcalpowera	pcalpower
	pcalpowerb	
Phase-cal spurious signals	pcalspur	pcalspur
Sampler statistics	samplestat	samplestat
Overnight stability	overnite	plotlog
Short-term stability	rapid	plotlog
Orientation effects	rapid	plotlog
Mark 5A tests	chk1024	chk1024
	vchk1024	

Table 1. Correspondence of Tests, SNAP procedures/schedules, and analysis scripts.

Individual scripts have special options that can be used to control how the data is processed and/or plotted. The "`-h`" option for each script (e.g., "`plotlog -h`") gives a general description of the features of the script and all its options.

Scripts that produce plots will display them in your graphical environment if one is defined. If not, they will by default place them in a file with a generic name using a portrait PostScript encoding. There are options available, the "`-f file/device`" option most notably, to specify he file name and control the way the graphics are generated. The "`-h`" option gives more information about the "`-f`" options in scripts where it is available. Please note that the plot device "`vps`" is generally the most useful for file output. Among other considerations, the resulting file can usually be printed with the Linux "`lpr`" command. For example:

```
pcalamp -f /tmp/pcalamp.eps/vps pcalmpsv.07.100.log
lpr /tmp/pcalamp.eps/vps
```

Please note that if you specify a "`file`" in the "`-f file/device`" option and fail to specify the "`/device`", a lot of Pgplot error messages are produced and no plot is generated.

Scripts that produce text can have their output sent to a file and/or printed using standard UNIX redirection techniques:

```
pcalpower pcalpowersv.07.100.log >/tmp/pcalpowersv.07.100.txt
lpr /tmp/pcalpowersv.07.100.log
```

or

```
pcalpower pcalpowersv.07.100.log | lpr
```

The output from the analysis scripts, whether a plot or test, includes the name of the station that the data came from, the name of the log analyzed, and the date the data was collected.

chk1024

This section describes some of the features of the “chk1024” analysis script for Mark5A recorders. The discussion here is applicable to the version in FS 9.9.3 and later. This script is intended to summarize the results of running the “chk1024” or “vchk1024” schedule. By default, the script will produce “overall” results for each setup (usually “4f”, “1g”, and “512”) combined over each mode (usually “idle” and “play”) for all the tests. The tests for each track are:

- “frame” for determining if a sufficient number of frames were processed to make the “error” test useful
- “error” for the data error rates,
- “aux” for correct auxiliary data.

In addition, if the scan_check command output was missing or bad for a given mode, an appropriate message is printed. If a set-up appears more than once, only the results from the last set-up are displayed. If no decoder firmware information is in the log or if an old version was used, the appropriate fact will be reported. Certain other known problems that can affect the results will also be reported if they occur.

More detail than shown in the default output can be obtained using the “-s” (summary) option. This option prints summary statistics for each test for each mode used with each setup. This can be useful for distinguishing which mode and/or test failed for a set-up if all did not. In addition, the combined results for each setup and mode combination are broken down by input “set” and odd/even tracks within each track. This is useful if the problem in a given mode was with one input “set” and/or odd/even tracks. The summary report lists “bad” versus “reported” tracks. The reported tracks may differ from the “total” (or “tested”) tracks because of decoder communication problems. Please note that maximum reported tracks in the “set” listings are limited by the subset of tracks included in each type.

There are several useful command line options that can be used to dissect the results. The “-b” or “-g” options can be used to get a listing of the bad or good, respectively, tracks in each set-up and mode combination for a given test: “aux”, “error”, or “frame”. The “-r” option can be used to get a print-out of the raw data from a given track in all tests. The “-t” option can be used to get a print-out of the “processed” data from a given track in all tests. Any combination of options “bgt” or “bgr” cannot be used simultaneously. Appropriate error messages are issued if they are.

Two options, “-a” and “-p”, can used to overcome certain problems with the data. These options should be used with care. They are useful for getting a better idea of what is going on with the data when there are certain problems. However, they should not be used in lieu of correcting the problems. The “-a” (aux) option can be used to removed the effect of bad auxiliary data from the rest of the results. This option is useful primarily if the formatter produced bad auxiliary data fields and/or if the auxiliary data is missing from the log because of decoder communication problems. The “-p” (permissive) option can be used to recover from

The “frame” test is meant to verify that a minimum number of frames, default 300, has been processed for evaluating the “error” tests. If for some reason this number wasn’t processed, but you still want to evaluate the results, you use the “-f” option to reduce the number of required frames. It is of course also possible to increase the number of required frames, but this may not be helpful unless the test schedule is revised.

plotlog

The “plotlog” analysis script is worth some additional discussion. This script is written as a general utility to plot ancillary data from a log file, including: wx, cable, gps-fmout, setcl, Tsys, receiver parameters, phase-cal amplitude, and phase-cal phase. Besides being useful for examining the results of the stability tests in the “systests” suite, it is also useful for examining experiment logs. It can be used to plot the ancillary data collected at the end of the experiment or progressively during an experiment to look for problems. A set of plots generated at the end of an experiment can serve as a summary hardcopy record of the ancillary data from the experiment for your files.

By default the “plotlog” script will make simple plots of any and all of the usual ancillary data found in the log. Plots of other data can be added to the default if desired; please contact Ed (Ed.Himwich@nasa.gov) with any requests. By using command line switches, it is also possible to select plots of phase-cal phase-differences within IF band passes (switches “-1anw”) and phase-cal amplitude versus phase plots (switch “-p”). If any of these switches are used, the normal plots are suppressed. There is also a switch, “-s”, that will allow data associated with a different command such as “tpi” to be plotted instead of “tsys” (note that this is for “tsys” only and not “tsys1”, “tsys2”, and “tsys3” commands that are in old style logs and which are also plotted normally by “plotlog” if present). As usual a full explanation of all options is available with the “-h” switch. The “-j” option can be used to normalize the phase-cal amplitudes by the system temperature per channel. This is particularly useful with the “-p” option to look for spurs. The "-g include_string" (and "-e exclude_string") option to include (and/or exclude) plots based on regular expression match with the plot titles are useful for getting only specific plots. There is sample “plotlog” output among the samples at the end of this section. Another example is included in the “Experiment pre-checks and Operations” course notes in this notebook.

Example Output

Example output from the scripts follows this page. Some interesting features of each are tersely described here or with the output.

pcalamp: USB spectra generally flat, although a few too high at 7 MHz. Bad rolloff in L10.

pcalpower: Pcalpower 2-3%, if2 missing

pcalspur: USB spurs generally <~ -30 dBc, except for -20 dBc U1 spurs at 2010 & 3010 kHz, and assorted S-band spurs at 5 MHz harmonics. Dashed line is detection limit, measured values (solid line connected points) are generally at the limit which means no detection.

pcalimage: OK except U5/L5 ~15 dB rejection. Dashed line is detection limit, measured values (solid line connected points) are generally at the limit which means no detection.

samplestat: U8 the worst, at +-6% deviation from ideal

plotlog: OK, rain last six hours? Phase-cal phase-differences within bands mostly flat, noisier last six hours (rain?); phase-cal amp versus phase (at 10 kHz for the six USBs sampled) shows minor spurs (single and/or double sinusoids) in some X channels, S phase range too limited to see much

chk1024: several sample outputs with introductory comments are listed

SYSTESTS - System test measurements and analysis package

Ed Himwich, Brian Corey, and Dan Smythe

September 3, 2008

INTRODUCTION

This document describes how to install and use the "systests" tools for measuring, plotting, analyzing system test data. The tests supported are: (1) band pass shapes, (2) baseband/video converter image rejection, (3) phase-cal power measurements, (4) phase-cal spurs, (5) sampler statistics, (6) "overnite" stability, (7) short-term stability, (8) orientation dependent effects, and (9) Mark 5 tests. Most of these tests require that you have a Mark IV Formatter and Decoder with either a VLBA or Mark IV rack. However, test (3) only requires a working "caltsys" procedure, and tests (6), (7), and (8) can be modified to do a more limited test for other configurations.

Once the tools are installed and set-up, each test requires only two type-ins, one to make the measurement and one to produce the plots or analysis results. Some tests require manual intervention by the operator to set switches or disconnect/reconnect cables. Everything else is automatic. The remainder of this document describes the usage and installation of the tools. The "help" output from the data processing scripts is provided for reference.

For stations that are upgrading to VSI4/Mark 5b configurations, a "systvsi4.prc" procedure library that will work with the VSI4 board is included in /usr2/fs/systests. An existing installed "systests.prc" library can be modified to support the VSI4 board by changing all strings "pcalports=" to "vsi4=,". No other changes are needed.

USAGE

With exception of the Mark 5 tests, after the tools have been installed, you can make the measurements for any observing mode. You should set up the rack in the mode you want to test it in before doing any tests. For example, to use the RD mode at Gc, you could use:

```
proc=rd0505gc
setup8f
```

After the mode has been set-up, you can make any or all of the measurements described below.

For Mark 5 tests, you need to use the chk1024.snp and chk1024.prc (for Mark IV, use vchk1024.snp and vchk1024.prc for VLBA4) files developed by Dan Smythe to make the measurements. This discussed in more detail below.

BANDPASS MEASUREMENTS

The amplitudes of all the phase cal tones within a baseband channel are measured and plotted to provide a power spectrum with coarse frequency resolution.

1. To collect data for bandpass plots, make sure that the equipment is set-up in the mode you want to test in (see above under USAGE), then enter:

   ```
   log=<pick a convenient name, maybe something like "testNN">
   proc=systests
   pcalamp
   ```

   For "NN" in the log file name, you can use a two digit number,

possibly sequential with other numbers used for other tests.

When you see the output "pcalamp done" in the log display, the measurement is complete. The procedure takes about two minutes to run.

2. Plot the results, from a shell prompt try:

```
cd /usr2/log
pcalamp testNN.log
```

This will make a graphical display, if you are on X server. If not, or if this is too big on your display, or you want hardcopy output for documentation, you can generate a printed version with:

```
cd /usr2/log
pcalamp -f pcalamp.ps/vps testNN.log
lpr pcalamp.ps
```

Other output formats are available besides PostScript, try:

```
pcalamp -f \?
```

for more information.

3. If you want to make additional measurements, you must open a new log for each measurement, e.g., after having modified the equipment. The script will only plot the first amplitude results it finds in the log.

IMAGE REJECTION MEASUREMENTS

The amplitudes of the phase cal tones at the normal and image frequencies in both the USB and LSB sidebands of each channel are measured to provide estimates of the image rejection as a function of frequency.

1. To collect data for image rejection plots, make sure that the equipment is set-up in the mode you want to test in (see above under USAGE), then enter:

```
log=<pick a convenient name, maybe something like "testNN">
proc=systests
pcalimage
```

For "NN" in the log file name, you can use a two digit number, possibly sequential with other numbers used for other tests.

When you see the output "pcalimage done" in the log display, the measurement is complete. The procedure takes about four minutes to run.

2. Plot the results, from a shell prompt try:

```
cd /usr2/log
pcalimage testNN.log
```

The lowest level of image rejection that can be measured, the "sensitivity", is shown as dotted line in the graphs. The sensitivity is calculated from the phase cal amplitudes in the sideband not under test. The image rejection values measured are shown as points connected by lines. The solid line may overlay the dotted line in many cases. If the points plotted are not above the dotted line, no image rejection problem was found above this level, i.e., it is below the level that could be detected in this configuration.

Other options for output are available as described above under "BANDPASS MEASUREMENTS".

3. If you want to make additional measurements, you must open a new log for each measurement, e.g., after having modified the equipment. The script will only plot the first amplitude results it finds in the log.

PHASE CAL POWER MEASUREMENTS

The fractional contribution of phase cal to the total system power in each channel is measured.

1. How you make this measurement would normally depend on whether your station has FS control to turn your phase-cal signal on and off. If you have computer control of your phase-cal, use procedure "pcalpowera". That procedure will alternate making measurements with the phase-cal on and off. If you don't have computer control use the "pcalpowerb" procedure, which will make several measurements at a time in each state to reduce the number of manual steps that are necessary.

   For automatic switching, make sure that the equipment is set-up in the mode you want to test in (see above under USAGE), then use these commands:

   ```
   log=<pick a convenient name, maybe something like "testNN">
   proc=systests
   pcalpowera
   ```

   For "NN" in the log file name, you can use a two digit number, possibly sequential with other numbers used for other tests.

   When you see the output "pcalpowera done" in the log display, the measurement is complete. The procedure should take about seven minutes to execute.

   For manual switching, make sure that the equipment is set-up in the mode you want to test in (see above under USAGE), then use these commands:

   ```
   log=<pick a convenient name, maybe something like "testNN">
   proc=systests
   pcalpowerb
   ```

   For "NN" in the log file name, you can use a two digit number, possibly sequential with other numbers used for other tests.

   When you see the output "pcalpowerb done" in the log display, the measurement is complete. The procedure should take about seven minutes to execute plus however long it takes you to respond to the prompts. There should be five prompts to respond to. After taking the requested action, click the "continue" button on the prompt window to continue.

   For manual switching, when prompted to turn the phase-cal on, you should make sure the system is set-up normally. This includes specifically that the phase-cal is on. When prompted to turn the phase-cal, you should turn the phase-cal off, preferably without directly disconnecting the 5 MHz that runs to the receiver for the phase-cal.

2. Plot the results, from a shell prompt try:

   ```
   cd /usr2/log
   pcalpower testNN.log
   ```

The average results for each detector in each state (pcal on and off) are printed followed by the differenced results including the fractional power due to the phase-cal signals. If you use the "-b" option of the script with the single-side band bandwidth of the video- or base-band converters (for example "pcalpower -b 16 testNN.log", where "16" is the channel bandwidth), the script will also print the approximate amplitude that should be seen for individual pcal tones in Mark IV decoder units. This is convenient for comparison with the bandpass measurements described above.

The "noisy" columns in the output contains an asterisk, "*", if the raw data for that device had a spread of six or more times its RMS scatter between the largest and smallest values. This gives some suggestion about whether the data are too noisy to be useful, but the presence or absence of an asterisk is no guarantee that the data are bad or good, respectively.

If you want to save the results in a file or print them, then you can redirect them to file, for example:

```
pcalpower testNN.log >pcalpower.txt
lpr pcalpower.txt
```

3. If you want to average additional measurements you can continue to take them in the original log. The "pcalpower" script averages all the values for each state (pcal on and off) that it finds in the log or logs specified on the command line. However, if want to make measurements to see the results of a hardware change you should start a new log.

PHASE CAL SPUR MEASUREMENTS

Evidence for spurious phase cal signals is sought by measuring the amount by which the phase cal level drops when phase cal is turned off and when the receiver LO is unlocked.

1. To collect data for phase cal spur plots, make sure that the equipment is set-up in the mode you want to test in (see above under USAGE), then enter:

```
log=<pick a convenient name, maybe something like "testNN">
proc=systests
pcalspur
```

For "NN" in the log file name, you can use a two digit number, possibly sequential with other numbers used for other tests.

When you see the output "pcalspur done" in the log display, the measurement is complete. The procedure should take about eight minutes to execute plus however long it takes you to respond to the prompts if any. There may be up to five prompts to respond to. When you complete each requested action, click the "Continue" button in the prompt window to continue.

If prompted to turn the phase-cal on, you should make sure the system is set-up normally. This includes specifically that the phase-cal is on and all LOs are locked. If prompted to turn off the phase-cal, you should turn the phase-cal off, preferably without physically disconnecting the 5 MHz reference that runs to the receiver for the phase-cal. (At some stations, the reference is at some frequency other than 5 MHz, such as 500 MHz.) For instance, if the 5 MHz comes from a Mark IV-style delay cal ground unit, use the front-panel switch on the ground unit to turn phase cal off. If prompted to unlock the LO, you should unlock the receiver LO, possibly by disconnecting the 5 MHz (or other reference frequency) to which the LO is locked, but not by actually turning the LO off; make sure the phase-cal is turned back on again

for this measurement. If you are prompted to turn off the phase-cal 5 MHz, you should lock the receiver LO again and then physically disconnect the cable that carries 5 MHz (or whatever the phase-cal reference frequency is) to the receiver for the phase-cal. If you had to disconnect the 5 MHz (or other reference frequency) when you turned off the phase-cal when prompted to do that, this is okay. However, you should understand that you will not get different measurements for the two cases. Lastly, if you are prompted to turn on the 5 MHz for the phase-cal, you should do that. For this last prompt the system should be restored to normal.

2. Plot the results, from a shell prompt try:

```
cd /usr2/log
pcalspur testNN.log
```

The lowest level of spur that can be measured, the "sensitivity", is shown as a dotted line in the graphs. The sensitivity is calculated from the phase cal amplitude when the phase cal is on. The spur levels are shown as points connected by lines. The solid line may overlay the dotted line in many cases. If the points plotted are not above the dotted line, no spur was found above this level. There may be spurs below the line. The "-s" option of the "pcalspur" script will generate an additional plot of just the sensitivity.

Other options for output are available as described above under "BANDPASS MEASUREMENTS".

3. If you want to make additional measurements, you must open a new log for each measurement, e.g., after modifying the equipment. The script will only plot the first spur results it finds in the log.

SAMPLER STATISTICS MEASUREMENTS

The sampler statistics for each channel are recorded for comparison to ideal values.

1. To collect data for sampler statistics plots, make sure that the equipment is set-up in the mode you want to test in (see above under USAGE), then enter:

```
log=<pick a convenient name, maybe something like "testNN">
proc=systests
samplestat
```

For "NN" in the log file name, you can use a two digit number, possibly sequential with other numbers used for other tests.

When you see the output "samplestat done" in the log display, the measurement is complete. The procedure should take about two minutes to execute..

2. Plot the results, from a shell prompt try:

```
cd /usr2/log
samplestat testNN.log
```

The deviation from ideal percentages for each level is displayed. The levels are "--" (V < -0.2v), "-" (-0.2v < V < 0v), "+" (0v < V < +0.2v) and "++" (+0.2v < V). The ideal levels are represented as dotted lines. The actual measurements are represented by dots with solid lines connecting them. The solid lines may overlay the dotted lines. The raw percentages can be displayed by using the "-r" command line switch for the script.

Other options for output are available as described above under "BANDPASS MEASUREMENTS".

3. If you want to make additional measurements, you must open a new log for each measurement, e.g., after having modified the equipment. The script will only plot the first statistics results it finds in the log.

OVERNITE STABILITY MEASUREMENTS

The values of many parameters are recorded over an extended period to look for stability problems.

1. To collect data for overnite stability plots, make sure that the equipment is set-up in the mode you want to test in (see above under USAGE), then enter:

```
log=<pick a convenient name, maybe something like "testNN">
proc=systests
overnite
```

For "NN" in the log file name, you can use a two digit number, possibly sequential with other numbers used for other tests.

You should let the procedure continue to make measurements as long as possible, preferably over a 24 hour or longer period, but at least overnight (more than 8 hours). When you want to stop the measurements, enter:

```
onsample@
log=station
```

2. Plot the results, from a shell prompt try:

```
cd /usr2/log
plotlog testNN.log
```

The data plotted depends on what is in the log. This will depend on what was sampled and whether there were errors that prevented some measurements from returning results. Any and all of the following items found in the log are plotted: wx/, cable/, rx/ (or sx/ or sk/), fmout-gps/ (or gps-fmout/), tsys/, and for a single tone per upper sideband channel: the pcal amplitude and phase measurements. The difference in phase between channels on the same IF can be plotted by selecting the appropriate one of the '1anw' command line switches (see "plotlog -h" for more information).

Other options for output are available as described above under "BANDPASS MEASUREMENTS".

3. If you want to make additional measurements, you would probably want to open a new log for each measurement. All the data from a single log are plotted together, regardless of the time span.

SHORT-TERM STABILITY MEASUREMENTS

The values of many parameters are recorded very rapidly for a short time to look for short term stability problems.

1. To collect data for short-term stability plots, make sure that the equipment is set-up in the mode you want to test in (see above under USAGE), then enter:

```
log=<pick a convenient name, maybe something like "testNN">
proc=systests
rapid
```

For "NN" in the log file name, you can use a two digit number, possibly sequential with other numbers used for other tests.

You should let the procedure continue to make measurements for a reasonable amount of time, perhaps 30 minutes. When you want to stop the measurements, enter:

```
stsample@
log=station
```

2. Plot the results, from a shell prompt try:

```
cd /usr2/log
plotlog testNN.log
```

The data plotted depends on what is in the log. This will depend on what was sampled and whether there were errors that prevented some measurements from returning results. Any and all of the following items found in the log are plotted: wx/, cable/, rx/ (or sx/ or sk/), fmout-gps/ (or gps-fmout/), tsys/, and for a single tone per upper sideband channel: the pcal amplitude and phase measurements. The difference in phase between channels on the same IF can be plotted by selecting the appropriate one of the '1anw' command line switches (see "plotlog -h" for more information).

Other options for output are available as described above under "BANDPASS MEASUREMENTS".

3. If you want to make additional measurements, you would probably want to open a new log for each measurement. All the data from a single log are plotted together, regardless of the time span.

ORIENTATION DEPENDENT STABILITY MEASUREMENTS

The values of many parameters are recorded very rapidly while antenna is moved around its range of motion to look for orientation dependent effects.

1. To collect data for orientation dependent plots, make sure that the equipment is set-up in the mode you want to test in (see above under USAGE), then enter:

```
log=<pick a convenient name, maybe something like "testNN">
proc=systests
rapid
```

For "NN" in the log file name, you can use a two digit number, possibly sequential with other numbers used for other tests.

You should proceed to move the antenna around its range of motion either manually or using appropriate "source=azel,..." commands. As the antenna is moved, especially if this being done manually, please enter comments to describe what is happening, e.g. "now moving to el 0 to 90 along az 0". It is desirable to at least travel along the perimeter of the antenna's range of motion, including the extremes of the cable wrap for AZEL antennas. Traveling along the entire range of at least one intermediate position for each axis is also be desirable. If the "source=azel,..." (or "source=xy,...") command works for your antenna and your antenna's motions is predictable enough, you may be able to write a schedule to automate the motion.

When you have finished moving the antenna, you can stop the measurements, by entering:

```
stsample@
log=station
```

2. Plot the results, from a shell prompt try:

```
cd /usr2/log
plotlog testNN.log
```

   The data plotted depends on what is in the log. This will depend on what was sampled and whether there were errors that prevented some measurements from returning results. Any and all of the following items found in the log are plotted: wx/, cable/, rx/ (or sx/ or sk/), fmout-gps/ (or gps-fmout/), tsys/, and for a single tone per upper sideband channel: the pcal amplitude and phase measurements. The difference in phase between channels on the same IF can be plotted by selecting the appropriate one of the '1anw' command line switches (see "plotlog -h" for more information).

   Other options for output are available as described above under "BANDPASS MEASUREMENTS".

3. If you want to make additional measurements, you would probably want to open a new log for each measurement. All the data from a single log are plotted together, regardless of the time span.

MARK 5 TESTS

The quality of data sent to Mark 5 is analyzed for both idle and play modes to look for problems. More than one set-up is used, including the most stressful, 1 Gb/second.

1. To collect data, use the test schedule that is appropriate for your station's equipment, vchk1024.snp or chk1024.snp. Other schedules with simialr structures, but using different set-ups can be used as well. Before running the schedule, you should rename any existing log with the schedule's name. For example, if you you are using chk1024.snp, then enter the shell commands:

```
cd /usr2/log
mv chk1024.log chk1024_yymmmdd.log
```

   Where "yymmmdd" is the date that the contents of the log were created. You could instead use some mnemonic string that will help identify the contents of the log.

   Then enter the SNAP commands:

```
schedule=chk1024,#1
```

   Wait for the schedule to complete, then close the log:

```
log=station
```

   Now would be a good time to rename the log as desribed above, before it is accidently appended to with other data. The suggestions for renaming described above can be used.

2. To analyze the results, enter the shell commands:

```
cd /usr2/log
chk1024 LOG
```

   where LOG is the name of the log file you want to analyze. The results for each set-up found in the log are summarized. The name of the analysis program is chk1024 regardless of whether your used the chk1024, vchk1024, or some other schedule. The program chk1024 has several options, try "chk1024 -h" for more information, of which the "-s" option is probably the most immediately useful, for example:

```
chk1024 -s LOG
```

INSTALLATION

The installation of these tools are fairly easy. The process is described in the INSTALLATION section for three possible scenarios: "first time installation", "update to include stability tests", and "update to include Mark 5 tests.".

FIRST TIME INSTALLATION

The steps to install for the first time are described below.

(1) Login as "oper".

(2) Check that you don't already have procedure libraries and SNAP schedules with the default names, enter:

```
ls /usr2/proc/systests.prc
ls /usr2/proc/*chk1024.prc
ls /usr2/sched/*chk1024.snp
```

If you do already have such files, you should either rename them or use different names in the next step for the target files to copy to. In the latter case, you will have to appropriately modify the "proc=..." and "schedule=..." commands in the USAGE examples above to open the correct library. It may also be that you already the current version of *chk1024.prc and/or *chk1024.snp; you can check this with "diff".

(3) Copy the default procedure libraries and schedules to "/usr2/proc" and "/usr2/sched", respectively and set their permissions and ownership. You should copy the "vchk1024.snp" and "vchk1024.prc" files instead of "chk1024.snp" and "chk1024.prc", if you have a VLBA4 rack instead of a Mark IV rack.

```
cd /usr2/proc
cp -a /usr2/fs/systests/systests.prc .
cp -a /usr2/fs/systests/chk1024.prc .          [or vchk1024.prc]
chmod a+w systests.prc chk1024.prc             [or vchk1024.prc]
chown oper.rtx systests.prc chk1024.prc        [or vchk1024.prc]

cd /usr2/sched
cp -a /usr2/fs/systests/chk1024.snp .          [or vchk1024.snp]
chmod a+w chk1024.snp                          [or vchk1024.snp]
chown oper.rtx chk1024.snp                     [or vchk1024.snp]
```

If you have a VSI4/Mark 5B equipment configuration, you should replace the line above:

```
cp -a /usr2/fs/systests/systests.prc .
```

with:

```
cp -a /usr2/fs/systests/systvsi4.prc systests.prc
```

(4) Customize the "systests.prc" (or whatever you named it) procedure library. You should use PFMED to make the changes. The procedures: "tpisamp", "bbcmanx", and "bbcagcx" are used only by the pcal power measurements. The "pcalonsys" and "pcaloffsys" procedures are used by both the pcal power measurements and the spur measurements. The "lounlocksys", "pcal5moffsys" and "pcal5monsys" procedures are used only for pcal spur measurements. The "onsample" and "stsample" procedures are used

for the stability tests.

The "tpisamp", "bbcmanx", and "bbcagx" procedures need to be modified depending on the type of rack: VLBA4 or Mark IV. There are comments in the procedures describing which lines should be uncommented for each case. In "tpisamp", the correct forms of the "tpi=..." and "tsys=..." commands must be uncommented. The ones that aren't to be used have to be commented out. For VLBA4 use the ones with "formbbc". For Mark IV uses the ones with "formvc". For station specific detectors use the ones with "u5". For the "bbcmanx" and "bbcagcx" procedures, the "bbcmanz" command and "bbcagcz" commands should be uncommented for VLBA4, but Mark IV should have them commented out.

The "pcalonsys", "pcaloffsys", "lounlocksys", "pcal5moffsys" and "pcal5monsys" procedures are used to control turning various signals on and off. Each is set-up by default to prompt the user to manually make the necessary changes. If your system can control any of these features, you can comment out the "sy=run fs.prompt ..." in the appropriate procedures and add commands where indicated to automate the control. The feature most likely to have automatic control at a station is turning the phase-cal on and off. You would edit the "pcalonsys" and "pcaloffsys" procedures to automate this for the tests. If possible, turning off the phase-cal should not be implemented by physically disconnecting the 5 MHz that runs the antenna for the phase-cal. However, if that is the only way to do it please understand that the phase-cal off and phase-cal 5 MHz disconnected measurements will not give different results. The "lounlocksys" procedure is intended to unlock the LO and turn the phase-cal on. Manually unlocking the LO can often be accomplished by removing the 5 MHz signal that it is locked too. It should not be accomplished by actually turning the LO off. The "pcal5moffsys" procedure should relock the LO and disconnect the phase-cal 5 MHz, as opposed to merely turning the phase-cal off. Lastly, the "pcal5monsys" procedure should restore the phase-cal 5 MHz, and the phase-cal and LO should be on when it ends.

The "onsample" and "stsample" procedures should be edited as described by their internal comments. The issues are: (1) whether or not you have a Mark IV decoder, (2) the name of the procedure or SNAP command to make fmout-gps (or gps-fmout) measurements, and (3) the name of the procedure that records all of the receiver monitor points.

(5) In any session (usually an xterm) that you intend to run the analysis scripts from, you will either need to logout and login again or type "rehash".

(6) Install pgperl if your computer doesn't already have it. Three cases are covered here: FS Linux 5 (FSL5), FSL6, and FSL7 (or later). If you have a different distribution installed, you will need to determine how to do this yourself. Ed may be able give you some advice if you need it. Instructions for the three cases covered here follow:

FSL5: Be sure that your "/etc/apt/sources.list" file does not mention "stable". All web references should explicitly use "woody". Since "woody" is no longer the "stable" distribution, you will need to to modify your web references to include the archive:

```
deb http://archive.debian.org woody main contrib non-free
deb http://non-us.debian.org woody/non-US main contrib non-free
deb http://security.debian.org woody/updates main contrib non-free
```

If you have any questions about this please contact Ed

before attempting the install. Once you have verified your "sources.list", as "root" please enter:

```
apt-get update
apt-get install pgperl
apt-get clean
```

FSL6: For these systems, it may still be possible to install pgperl in the normal way. First, be sure that your "/etc/apt/sources.list" file does not mention "stable". All web references should explicitly use "sarge". Then please try:

```
apt-get update
apt-get install pgperl
apt-get clean
```

If that does not work, try the directions above for FSL5 for accessing the archive, but use "sarge" in place of "woody" for the web references. If that also doesn't work, then a third alternative might be the procedure described for FSL7 below.

FSL7: For these (and later) systems, first be sure that your "/etc/apt/sources.list" file does not mention "stable". All web references should explicitly use "etch" for FLS7 or another name as appropriate for later systems. Once that is verified, enter the following commands as "root" (using cut-and-paste in a shell window will reduce the chances of making typing mistakes):

```
cd /root
apt-get update
aptitude install pgplot5
wget
http://http.us.debian.org/debian/pool/main/libp/libpng/libpng10-0_
1.0.18-1_i386.deb
wget
http://http.us.debian.org/debian/pool/contrib/p/pgperl/pgperl_2.18
-5 i386.deb
dpkg -i libpng10-0 1.0.18-1 i386.deb
dpkg -i pgperl_2.18-5_i386.deb
apt-get clean
```

Over time these ".deb" files will probably move to "archive.debian.org", so you may need to retrieve them from there instead.

(7) Make sure that your print command will print PostScript files. This should already be set-up for FS Linux 5 systems with the command "lpr". If you can't print PostScript from your FS PC, you can always transfer the PostScript file to another machine that can or use a different output format (see step 2 above under "BANDPASS MEASUREMENTS" above in the "USAGE" section) that your printer supports.

(8) Installation of the vchk1024/chk1024 .snp and .prc files are straightforward. If you already have the appropriate one for your configurations installed, you can skip this step.

UPDATE TO INCLUDE STABILITY TESTS

The steps to update your installation to include stability tests are described below. You can tell if the stability tests have already been added by whether the procedures "stsample" and "onsample" are already in your sysystests.prc library.

(1) Login as "oper".

(2) Copy the procedure library holding the new procedures to "/usr2/proc"

```
cd /usr2/proc
cp -a /usr2/fs/systests/systest2.prc .
chmod a+rw systest2.prc
```

If you have a VSI4/Mark 5B equipment configuration, you should replace the line above:

```
cp -a /usr2/fs/systests/systest2.prc .
```

with:

```
cp -a /usr2/fs/systests/systvsi42.prc systest2.prc
```

(3) Customize the "systest2.prc" library on /usr2/proc. You should use PFMED to make the changes. Two procedures need to be edited and are both located at the start of the file: "onsample" and "stsample". They should be edited as described by their internal comments. The issues are: (1) whether or not you have a Mark IV decoder, (2) the name of the procedure or SNAP command to make fmout-gps (or gps-fmout) measurements, and (3) the name of the procedure that records all of the receiver monitor points.

(4) Append the new procedures to the existing "systests.prc" library (in "/usr2/proc"). The FS should not have the library open when this is done. In fact, it is just safer to terminate the FS while doing this step. Once it is safe to proceed, the procedures can be added with:

```
cd /usr2/proc
cat systest2.prc >>systests.prc
```

If you named your library something other "systests.prc", you should use that name instead in the command above.

(5) Once step (4) is completed, you can delete the "systest2.prc" library you modified:

```
cd /usr2/proc
rm systest2.prc
```

(6) In any session (usually an xterm) that you intend to run the analysis scripts from, you will either need to logout and login again or type "rehash".

UPDATE TO INCLUDE MARK 5 TESTS

The steps to update your installation to include stability tests are described below. You can tell if the Mark 5 tests have been added by whether the chk1024.snp or vchl1024.snp SNAP schedule is in your /usr2/sched directory.

(1) Login as "oper".

(2) Check that you don't already have procedure libraries and SNAP schedules with the default names, enter:

```
ls /usr2/proc/*chk1024.prc
ls /usr2/sched/*chk1024.snp
```

If you do already have such files, you should either rename them or use different names in the next step as the target file to copy to. In the latter case, you will have to appropriately modify the "schedule=..." commands in the USAGE examples above to open the

correct library. It may also be that you already the current version of *chk1024.prc and/or *chk1024.snp; you can check this with "diff".

(3) Copy the default procedure libraries and schedules to "/usr2/proc" and "/usr2/sched", respectively and set their permissions and ownership. You should copy the "vchk1024.snp" and "vchk1024.prc" files instead of "chk1024.snp" and "chk1024.prc", if you have a VLBA4 rack instead of a Mark IV rack.

```
cd /usr2/proc
cp -a /usr2/fs/systests/chk1024.prc .          [or vchk1024.prc]
chmod a+w chk1024.prc                          [or vchk1024.prc]
chown oper.rtx chk1024.prc                     [or vchk1024.prc]

cd /usr2/sched
cp -a /usr2/fs/systests/chk1024.snp .          [or vchk1024.snp]
chmod a+w chk1024.snp                          [or vchk1024.snp]
chown oper.rtx chk1024.snp                     [or vchk1024.snp]
```

HELP OUTPUT

For reference the "help" output from each of the three scripts, "pcalamp", "pcalimage", "pcalpower", "pcalspur", "samplestat", "plotlog", and "chk1024" (listed last) is listed below.

PCALAMP

```
Usage: pcalamp  -c string -vhpi -u upper -l lower -f file/device logs
Synopsis: extracts and plots pcal amplitudes from log files

This script processes log files to produce plots of pcal amplitudes. The
default is pcal amplitudes for the first set of amplitudes that appear in the
file. This behavior can be modified with the '-c' option to plot the first
occurrence of amplitude data after a FS command that contains the specified
string.

Option explanations:
 -c string plot first set of amplitudes found after a command/procedure
    matching 'string' was recorded in the log, for this to work; the
    command/procedure matching 'string' must not be called from a procedure,
    unless the procedure is logged with 'xlog=on'
 -p plot using power units for comparison to autocorrelation plots
 -i autoscale each plot individually
 -l lower  eliminate points smaller than lower when auto-scaling plots
 -u upper  eliminate points larger  than upper when auto-scaling plots
 -v print program and PGPLOT version information and stop
 -h print this help information and stop
 -f file/device  send graphs to 'file' using PGPLOT 'device'
    if -f omitted and DISPLAY is defined, an xterm will be used
    if -f omitted and DISPLAY is not defined, 'pcalamp.ps/vps' will be used
    if '-f ?', you will be prompted for standard pgplot devices,
                be sure to quote '?', like \?
    'vps' (portrait PostScript) is a useful choice for 'device' for file
    output
```

PCALIMAGE

```
Usage: pcalimage -vhf file/device
Synopsis: extracts and plots image reject using pcal tones from log files

This script processes log files to produce plots of image rejection. The
lowest
image rejection that can be detected is plotted as a dotted line, which may be
overlaid by the actual results.
```

```
Option explanations:
 -v print program and PGPLOT version information and stop
 -h print this help information and stop
 -f file/device send graphs to 'file' using PGPLOT 'device'
    if -f omitted and DISPLAY is defined, an xterm will be used
    if -f omitted and DISPLAY is not defined, 'pcalimage.ps/vps' will be used
    if '-f ?', you will be prompted for standard pgplot devices,
               be sure to quote '?', like \?
    'vps' (portrait PostScript) is a useful choice for 'device' for file
    output
```

PCALPOWER

```
Usage: pcalpower -bhv vc_bandwidth logs
Synopsis: tabulates pcal percentage of tsys data

This script will collect all tsys data in a log with pcal on and pcal off and
calculate the fractional power due to the pcal. The '-b' option can be used
to print the results in the Mark IV Decoders units for the (voltage) amplitude
of a single tone.

Whether the phase-cal is on or off for particular tsys measurements is
determined by whether the string 'pcalon' or 'pcaloff' was most recently
encountered in the log. Note that if 'pcalon' or 'pcaloff' is called from
a SNAP procedure, then it must be logged with 'xlog=on'.

Options explanation:
 -b vc bandwidth (e.g., 2, 4, 8, 16) for single tone voltage (Mk4 decoder
 units)
 -h print this help information and stop
 -v print program and PGPLOT version information and stop
```

PCALSPUR

```
Usage: pcalspur -svhf file/device logs
Synopsis: extracts pcal data from logs and plots pcal spurs

The level of spurs is plotted for each of three states that may occur:
pcaloff, lounlock, and pcal5moff. These states are plotted relative to the
'pcalon' state. Data for each of these cases in the log must follow logged
commands that contain these strings. If the commands are called from
procedures, they must be logged with 'xlog=on'.

Options explanation:
 -s make sensitivity (only) plot also
 -v print program and PGPLOT version information and stop
 -h print this help information and stop
 -f file/device send graphs to 'file' using PGPLOT 'device'
    if -f omitted and DISPLAY is defined, an xterm will be used
    if -f omitted and DISPLAY is not defined, 'pcalspur.ps/vps' will be used
    if '-f ?', you will be prompted for standard pgplot devices,
               be sure to quote '?', like \?
    'vps' (portrait PostScript) is a useful choice for 'device' for file
    output
```

SAMPLESTAT

```
Usage: samplestat -dvhf file/device logs
Synopsis: extracts and plots sampler statistics from log files

This script processes log files to produce plots of sample statistics. The
ideal values are plotted as a dotted line. The measured statistics appear
as points connected by lines.  The default mode is print the deviation from
```

ideal. The '-r' switch can used to plot the raw statistics instead.

Option explanations:
 -r plot raw statistics instead of deviation from ideal
 -v print program and PGPLOT version information
 -h print this help information and stop
 -f file/device send graphs to 'file' using PGPLOT 'device'
    if -f omitted and DISPLAY is defined, an xterm will be used
    if -f omitted and DISPLAY is not defined, 'samplestat.ps/vps' will be used
    if '-f ?', you will be prompted for standard pgplot devices,
               be sure to quote '?', like \?
    'vps' (portrait PostScript) is a useful choice for 'device' for file
    output

PLOTLOG

Usage:
   plotlog [options] logs
Synopsis: extracts and plots data from log files

This script finds wx/, cable/, tsys/ rx/ (or sk/ or sx/), gps-fmout/ (or fmout-gps/), setcl#time/, and the first encountered USB tone per channel for Mark IV decoder pcal data in log files and generates time plots of the data. The time axis of all the plots is the same. The data is plotted so that it fits
entirely inside the boxes. Points that appear along the top line of a plot did not decode ($$$ for example) or were edited out by command line options.

The '1anw' switches can be used to select different IF arrangements for calculating phase differences between pcal tones. Any combination of the '1anw'
switches can be used. No difference is plotted more than once.

The 'j' switch can be used to normalize pcal amplitudes by Tsys (per channel) for time plots and amplitude versus phase plots. The minimum non-zero Tsys (per channel) is used as the reference value for normalization. If this causes problems, an explict global reference value can be specified with the 'k' option. The 't' option may also be useful for removing bad Tsys values. By default dual side-band Tsys is used for normalization if it is available, if not then single side-band Tsys data is used. This is usually what is wanted but if not, the 'b' option can be used to force use of single side-band Tsys for channels that it is available for.

For the 'j1anw' switches a simple model is used to match different data from the log. Points from different data types that are the nearest together in time
are matched. No epoch from one data type is matched to more than one epoch of another type in this way. If the resulting matched epochs are within 60 seconds
of each other, the match is accepted. The tolerance of match can be adjusted using the 'u' option.

The 'p' switch adds amplitude versus phase plots for each phase-cal tone after all other plots. This option suppresses all other output except options -1anw.

Cable delay is plotted as one-way cable delay change unless the 'r' switch is used.

Option explanations:
 phase-difference plot options (suppresses all other output, except -p):
  -1 all converters are on one IF
  -a odd converters are on one IF, even are on another (astronomy)
  -n converters 1-8 on one band, 9-16 on a second (narrow band geodesy),
     also useful for wide band geodesy
  -w converters 1-4 on one band, 5-8 on a second, 9-16 on a third
     (wide band geodesy)

```
 -b        force single-side-band Tsys normalization of pcal amp data
 -c value  cable cal edit, if greater than zero, delete points that much or
           more above lowest value, if less than zero, delete points that much
           or more below largest value; uses plot Y axis units, be careful of
           the -r option
 -e regexp include only plots with labels that DO NOT match 'regexp', where
           'regexp' is any Perl regular expression pattern, you will need to
           quote 'regexp' if it contains special characters, e.g., 'Temp|Hum'
           or spaces, this option can be combined with the 'g' option as well.
 -g regexp include only plots with labels that DO match 'regexp', where
           'regexp' is any Perl regular expression pattern, you will need to
           quote 'regexp' if it contains special characters, e.g., 'Temp|Hum'
           or spaces, this option can be combined with the 'e' option as well.
 -j        normalize phase-cal amp by Tsys per channel
 -k value  changes 'j' normalization from minimum non-zero Tsys per channel to
           'value' for all channels
 -p        plot amplitude versus phase for phase cal tones,
           suppresses all other output except -1anw
 -r        plot cable data in raw units
 -s string match commands with 'string' instead of 'tsys',
           option -t still applies for editing
 -t value  edit out tsys points less than or equal to zero or greater than or
           equal to 'value'
 -z value  edit out pcal data with amplitudes less than 'value'
 -u value  maximum distance in time (seconds) allowed for matching data for
           '1anwp' option plots is changed to 'value' instead of 60 seconds
 -v        print program and PGPLOT version information and stop
 -h        print this help information and stop
 -f file/device  send graphs to 'file' using PGPLOT 'device'
    if -f omitted and DISPLAY is defined, an xterm will be used
    if -f omitted and DISPLAY is not defined, 'plotlog.ps/vps' will be used
    if '-f ?', you will be prompted for standard pgplot devices,
               be sure to quote '?', like \?
    'vps' (portrait PostScript) is a useful choice for 'device' for file
    output
```

CHK1024

```
Usage: chk1024 -acdhlpsuv -b a|e|f -f frames -g a|e|f -r track -t track logs
Synopsis: analyzes results of Mark5 check schedule v/chk1024 logs
```

This script will summarize the results of v/chk1024 or other similar Mark5 test schedule. The modes are determined from presence of 'setupX' procedure calls in the log. Where 'X' is a string. The mode is either 'play' or 'idle' depending on whether or not 'recscan' has been called since the last 'setupX'.

By default, chk1024 prints only the overall summary results grouped by setup. The number of tracks checked reflect those that were found in the log to have been selected with the Mark5.

The checks for each track are:

- Aux: considered good if decoded track number is correct
- Errors: considered good if synch, resynch, parity and CRC errors were zero
- Frames: considered good if more than the minimum frames were processed, minimum frames is settable, but defaults to 300 (0x12c)

Additional detail can be displayed by selecting command line options. The next level of detail is selected by the -s option, which will print a table of results for each check by setup and mode, also summarizing them by 'set' and odd/even track number. The 'reported' tracks are those for which the corresponding decoder output appears. If the 'Total' and 'reported' tracks don't agree this usually means there were decoder communication errors. The next level is shown with the -t option which shows the processed data for the selected track. Finally, the -r option will print the raw log records for the

selected track.

Please note that the set (odd/even) statistics and summary results at the end only report the total number of tracks found in the log, not the total expected. The user must verify that total tracks are correct for each setup. The 'Total Tracks' in the summary table is typically a good measure of the expected number of tracks, but they are distributed over the sets.

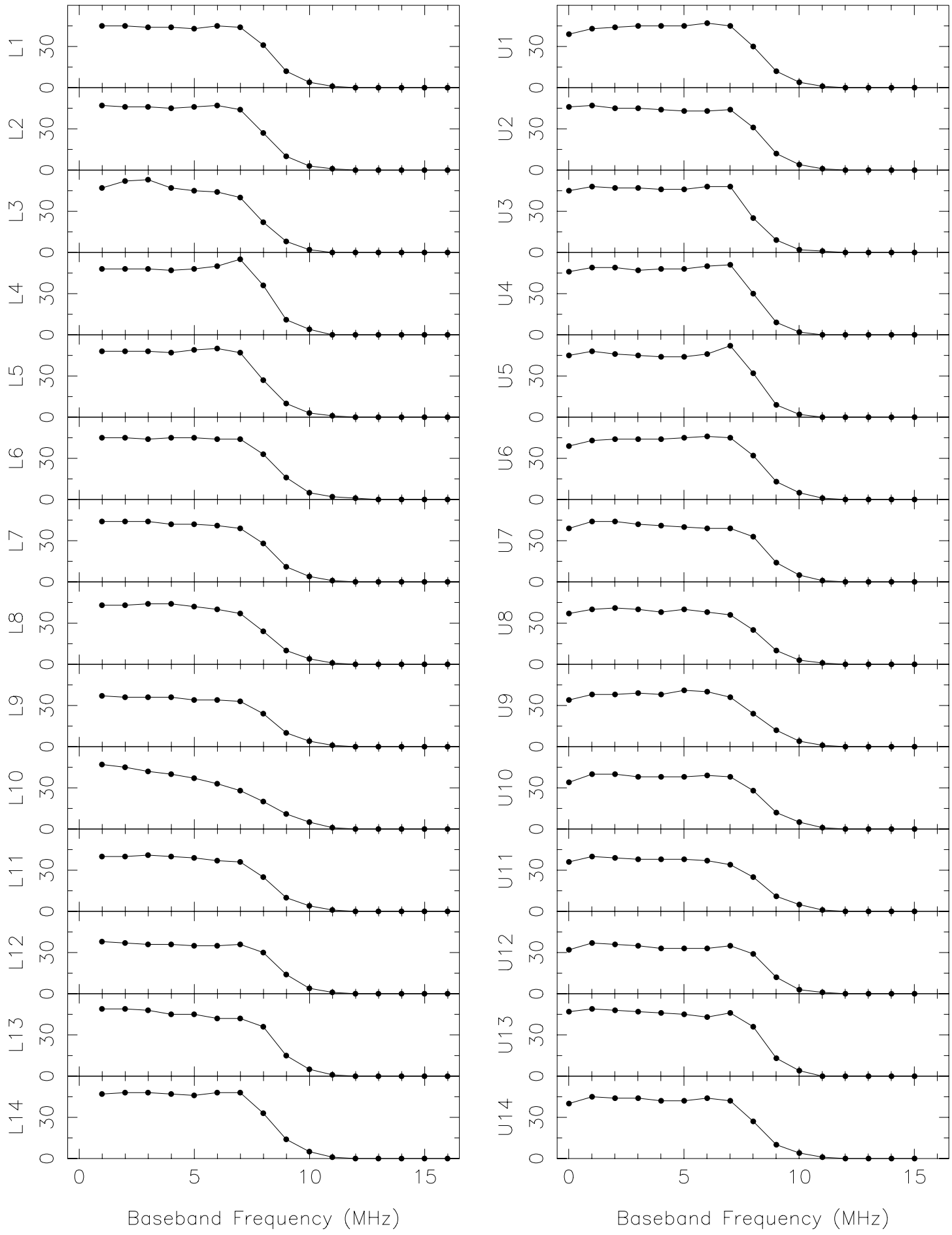
If the scan_check fails for a setup the phrase 'ERROR: scan_check failed' is included in the overall summary for the affected setup.

If the decoder firmware version is wrong or could not be determined a message to that effect is printed.

If the same set-up occurs more than once, results from all but the last are discarded. Note also that the 'Location' title line is printed once for each time it is encountered in the log, with the corresponding log time stamp. Typically, this happens for each time the schedule is started. Results from previous runs are discarded only for set-ups with the same names. All unique set-ups in the log are reported, even if they are from different runs. For a given set-up and mode if a track or scan_check appears more than once, the results for the last occurance is reported. A warning about 'Duplicate DQA records' indicates either a track (or tracks) was checked more than once or there was a decoder communication problem

Options explanation:

```
 -a         ignore aux data when calculating performance
 -b a|e|f   display  bad tracks for one of the Aux|Errors|Frames
            checks, cannot be used with -g, -t, or -r
 -c         show raw scan check data
 -d         show raw decoder firmware version data
 -f frames  sets the minimum level of required frames, default is 300 (0x12c)
 -g a|e|f   display good tracks for one if the Aux|Errors|Frames
            checks, cannot be used with -b, -t, or -r
 -h         print this help information and stop
 -l         lists each set-up as it is encountered, useful when a log contains
            more than one run, possibly on different dates
 -p         'permissive', handles some Mark IV decoder old firmware
            communication errors
 -r track   to display raw data for track, cannot be used with -b or -g
 -s         print summary statistics
 -t track   to display processed data for track, cannot be used with -b or -g
 -u         print tracks with duplicate dqa records
 -v         print program version information and stop
```

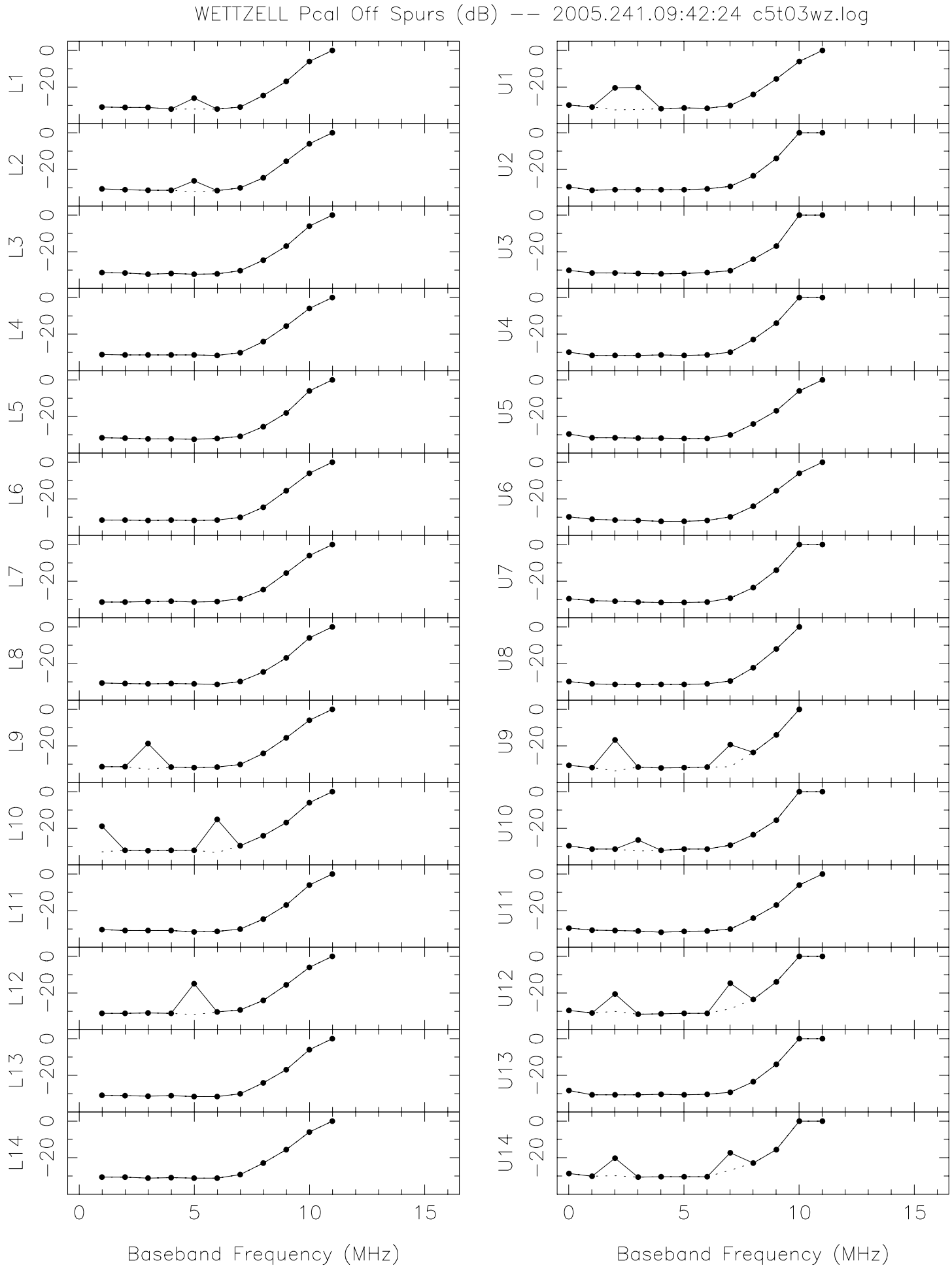

KOKEE Pcal Amplitude (Voltage, 0.1%) -- 2005.221.22:23:28 c5t01kk.log
L1
L2
L3
L4
L5
L6
L7
L8
L9
L10
L11
L12
L13
L14
U1
U2
U3
U4
U5
U6
U7
U8
U9
U10
U11
U12
U13
U14
0
30
0
5
10
15
Baseband Frequency (MHz)
Baseband Frequency (MHz)

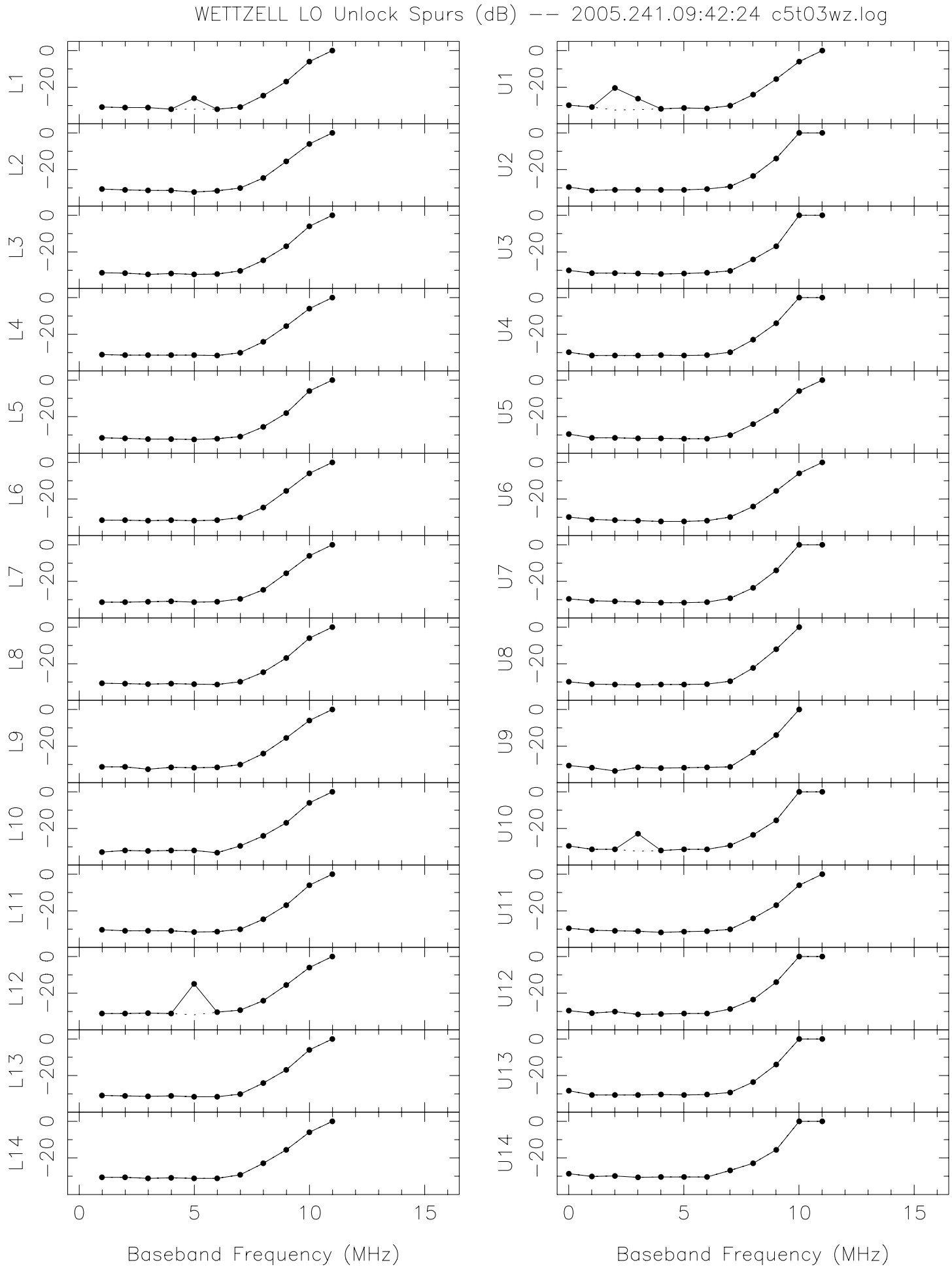
HartRAO pcal power -- 2005.209.08:27:58 c5t02hh.log

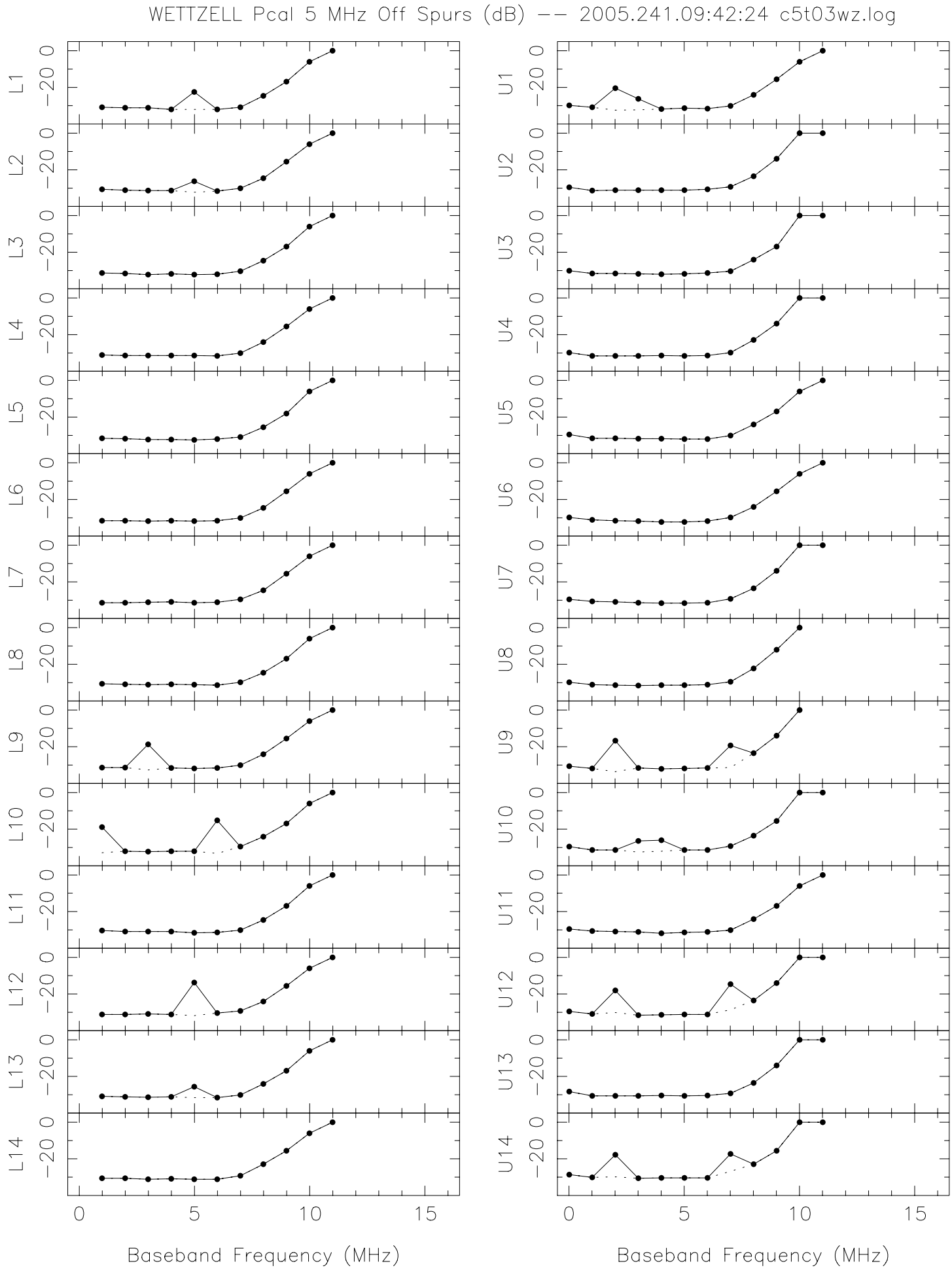
Ch	off	+/-	rms	n	max	min	(max-min)/rms	
1d	65.89	0.02	0.14	40	66.10	65.60	3.70	
2u	62.74	0.03	0.20	40	63.00	62.40	2.95	
3u	51.49	0.01	0.04	40	51.60	51.40	5.64	
4u	52.89	0.01	0.03	40	53.00	52.80	6.24	noisy
5u	51.60	0.01	0.04	40	51.70	51.50	5.03	
6u	55.46	0.01	0.06	40	55.60	55.40	3.62	
7u	58.30	0.01	0.07	40	58.40	58.20	2.91	
8d	59.13	0.01	0.09	40	59.30	59.00	3.43	
9u	45.63	0.01	0.05	40	45.70	45.60	2.13	
au	46.53	0.01	0.05	40	46.60	46.50	2.08	
bu	44.26	0.01	0.05	40	44.30	44.20	2.04	
cu	41.55	0.01	0.05	40	41.60	41.50	1.95	
du	42.50	0.00	0.02	40	42.60	42.40	8.72	noisy
eu	43.43	0.01	0.05	40	43.50	43.40	2.18	
i1	52.64	0.01	0.05	40	52.70	52.60	1.96	
i2								
i3	53.49	0.01	0.03	40	53.50	53.40	2.95	

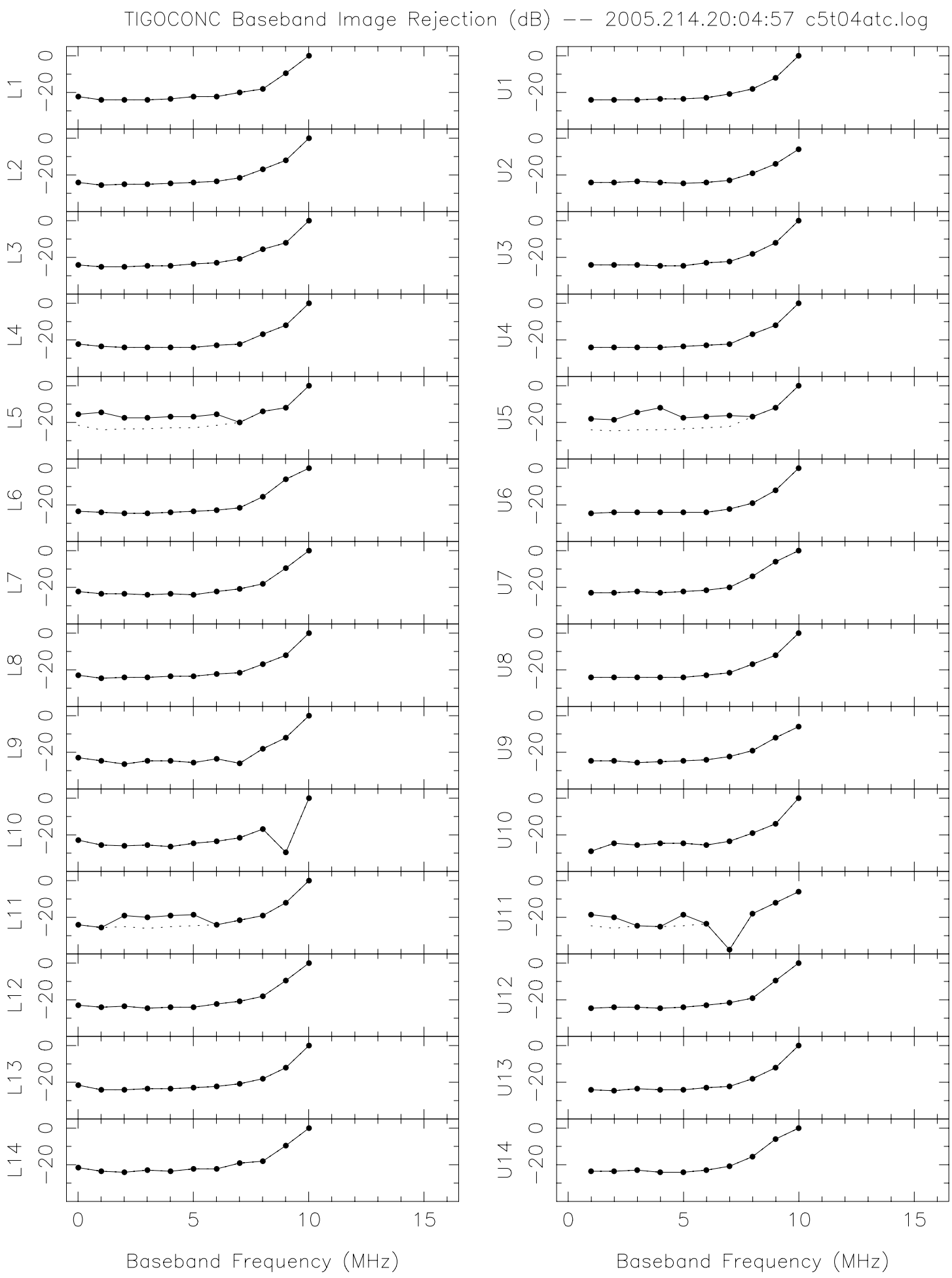
Ch	on	+/-	rms	n	max	min	(max-min)/rms
1d	67.28	0.02	0.11	41	67.50	67.10	3.61
2u	64.08	0.02	0.15	41	64.40	63.80	3.97
3u	52.65	0.01	0.05	41	52.70	52.60	1.95
4u	54.38	0.01	0.04	41	54.40	54.30	2.46
5u	53.06	0.01	0.06	41	53.20	53.00	3.59
6u	56.87	0.01	0.05	41	57.00	56.80	3.73
7u	59.71	0.01	0.03	41	59.80	59.70	2.98
8d	60.28	0.02	0.15	41	60.50	59.90	4.03
9u	46.72	0.01	0.06	41	46.80	46.60	3.17
au	47.63	0.01	0.07	41	47.70	47.50	2.88
bu	45.24	0.01	0.05	41	45.30	45.20	2.03
cu	42.56	0.01	0.08	41	42.70	42.50	2.58
du	43.32	0.01	0.08	41	43.50	43.20	3.69
eu	44.30	0.01	0.09	41	44.40	44.20	2.28
i1	53.97	0.01	0.05	41	54.00	53.90	2.06
i2							
i3	54.80	0.00	0.02	41	54.80	54.70	4.53

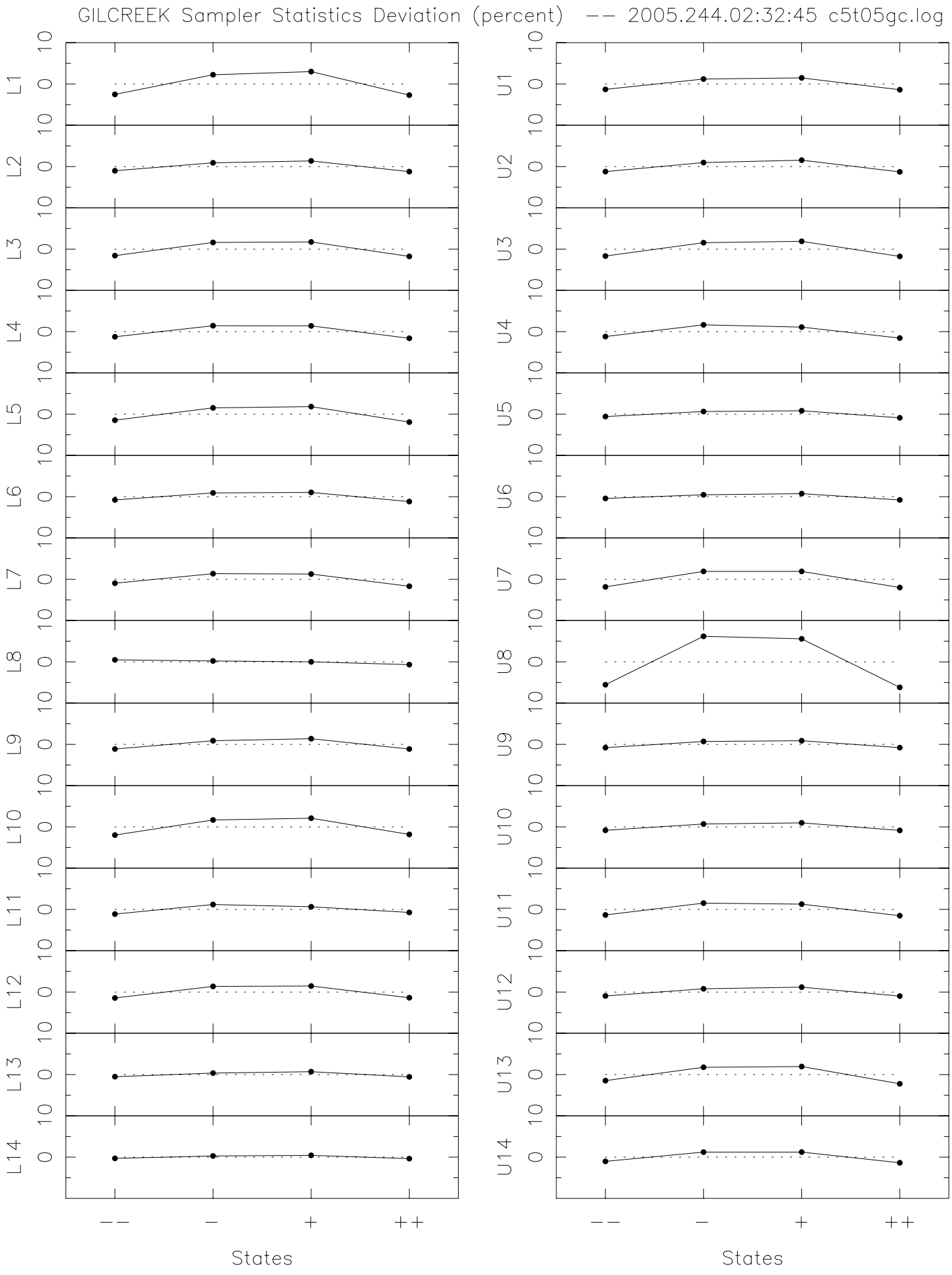
Ch	Diff	+/-	%	+/-%	
1d	1.39	0.03	2.11	0.04	
2u	1.34	0.04	2.13	0.06	
3u	1.16	0.01	2.25	0.02	
4u	1.49	0.01	2.81	0.02	noisy
5u	1.46	0.01	2.82	0.02	
6u	1.41	0.01	2.53	0.02	
7u	1.41	0.01	2.41	0.02	
8d	1.15	0.03	1.94	0.05	
9u	1.09	0.01	2.40	0.03	
au	1.10	0.01	2.36	0.03	
bu	0.97	0.01	2.19	0.02	
cu	1.01	0.01	2.43	0.04	
du	0.82	0.01	1.92	0.03	noisy
eu	0.87	0.02	2.01	0.04	
i1	1.32	0.01	2.51	0.02	
i2					
i3	1.31	0.01	2.44	0.01	

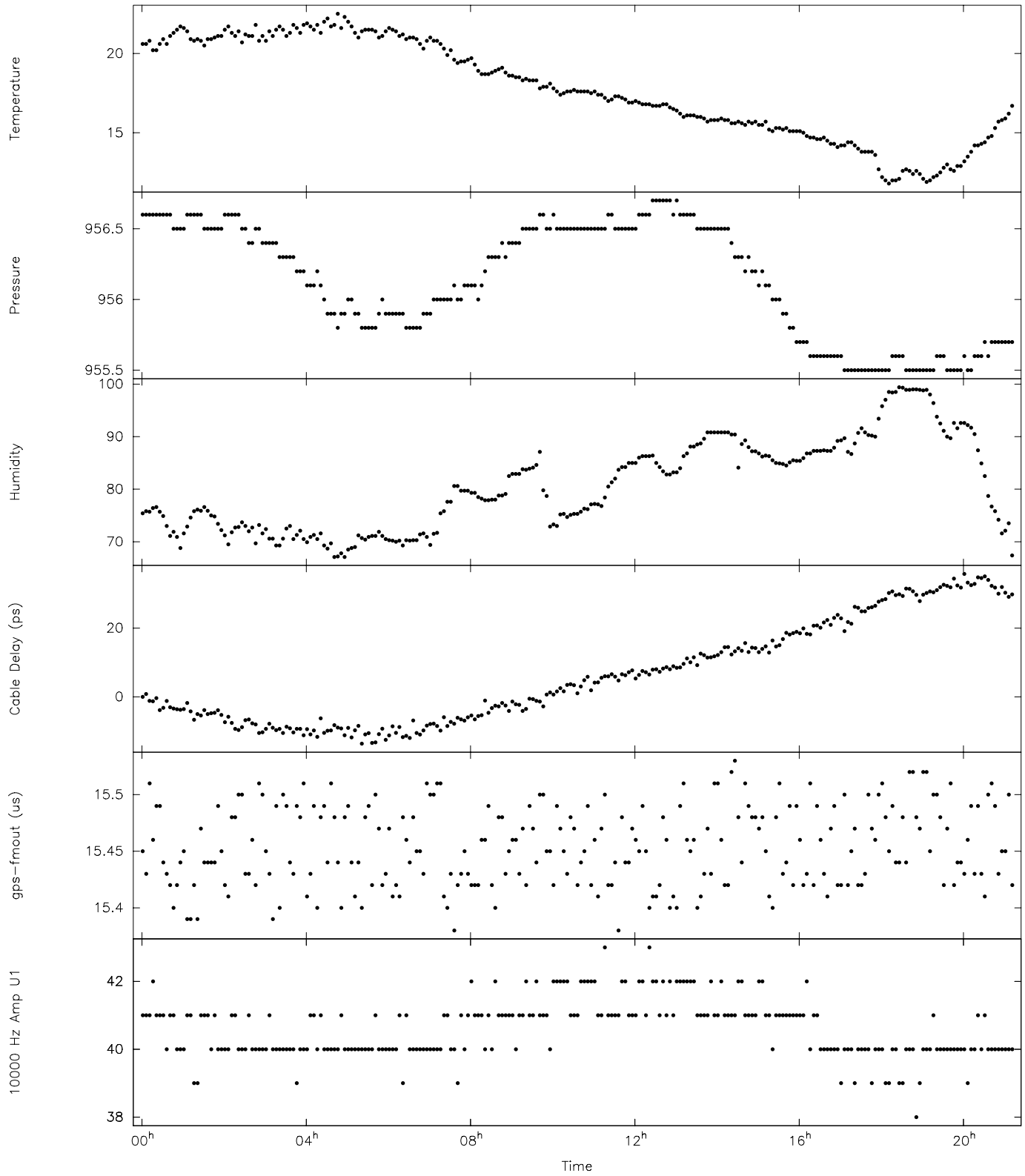

WETTZELL Pcal Off Spurs (dB) -- 2005.241.09:42:24 c5t03wz.log
L1
L2
L3
L4
L5
L6
L7
L8
L9
L10
L11
L12
L13
L14
U1
U2
U3
U4
U5
U6
U7
U8
U9
U10
U11
U12
U13
U14
-20 0
0
5
10
15
Baseband Frequency (MHz)
Baseband Frequency (MHz)

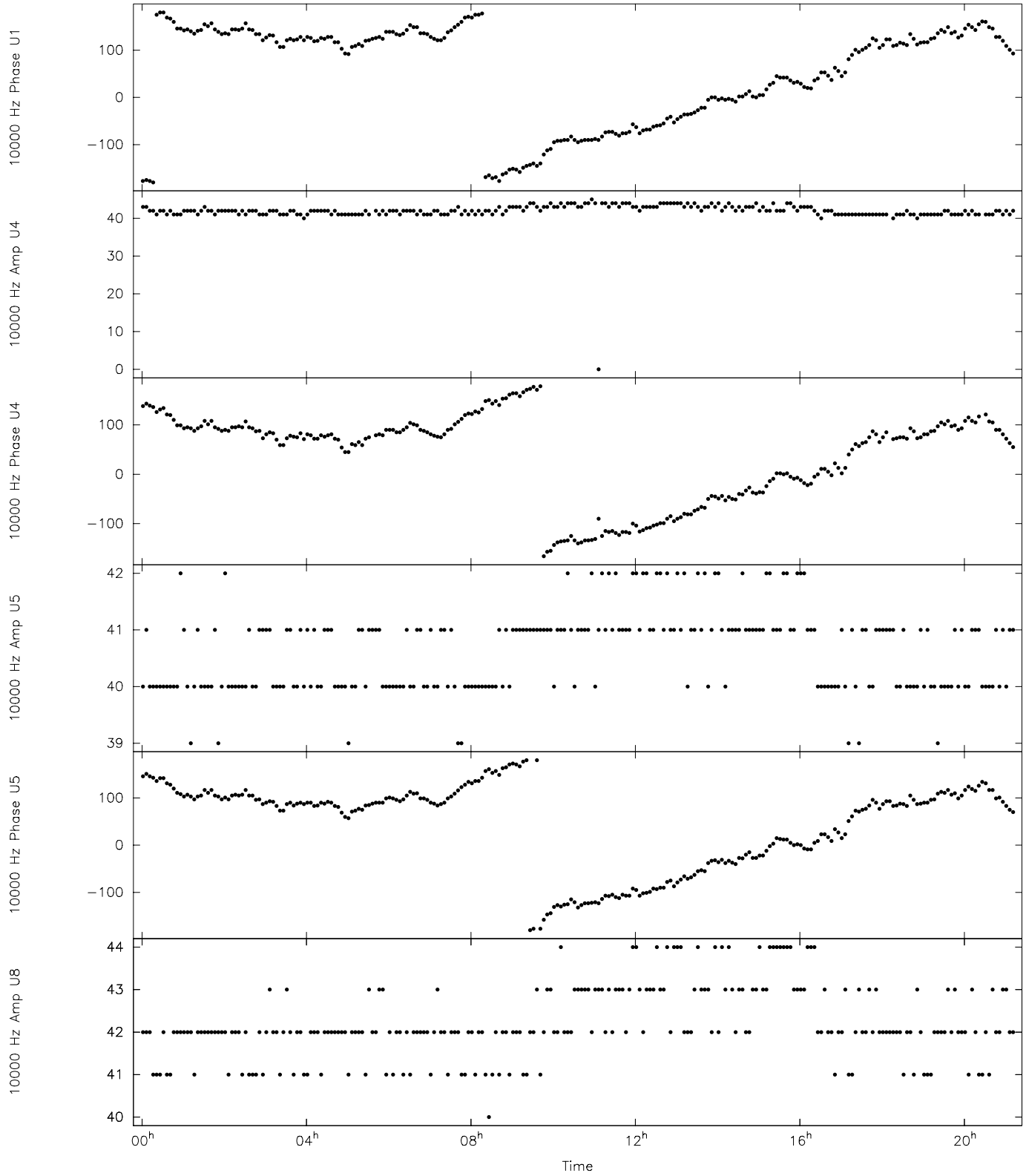

WETTZELL LO Unlock Spurs (dB) -- 2005.241.09:42:24 c5t03wz.log
L1
L2
L3
L4
L5
L6
L7
L8
L9
L10
L11
L12
L13
L14
U1
U2
U3
U4
U5
U6
U7
U8
U9
U10
U11
U12
U13
U14
-20 0
0
5
10
15
Baseband Frequency (MHz)
Baseband Frequency (MHz)

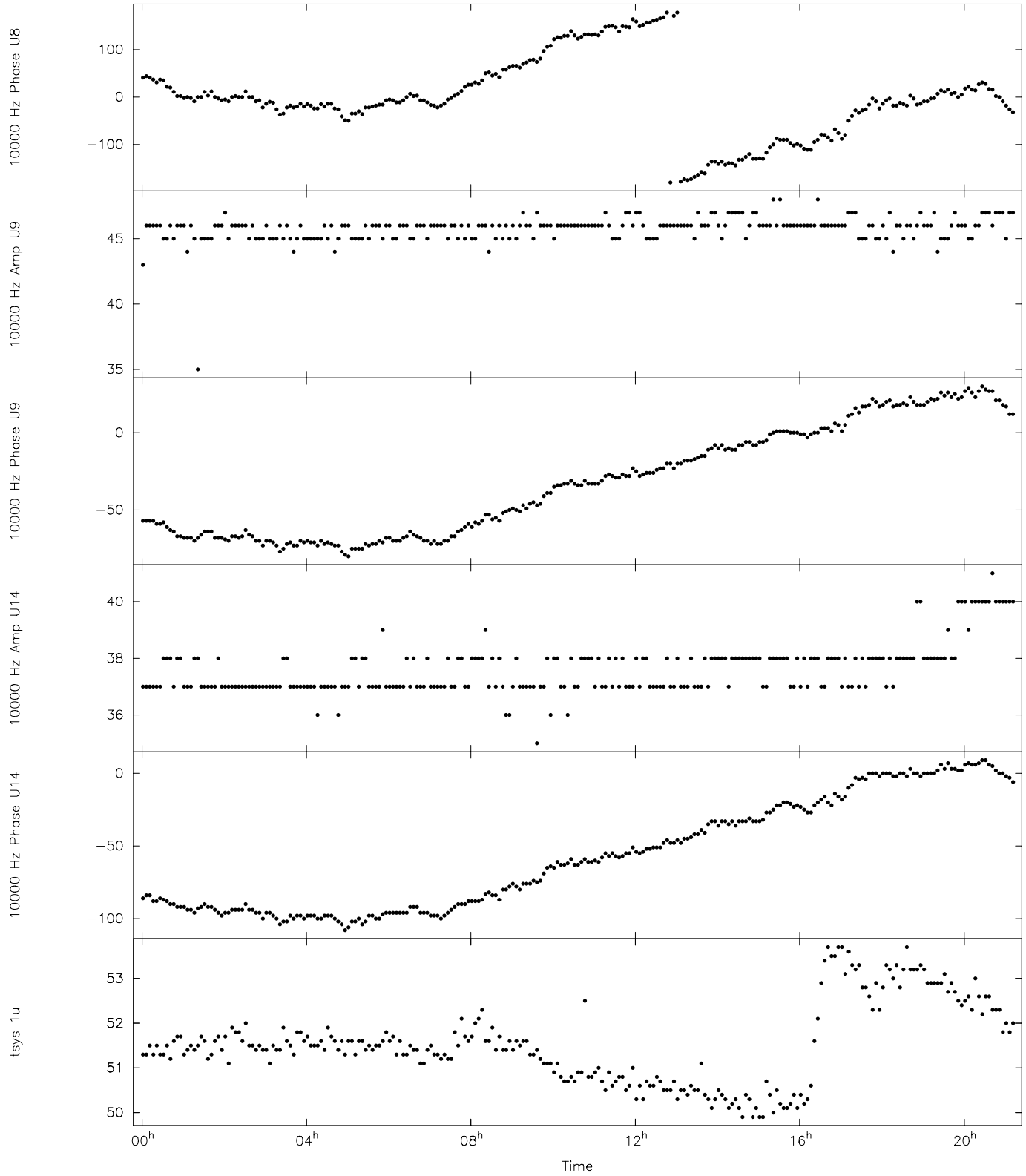

WETTZELL Pcal 5 MHz Off Spurs (dB) -- 2005.241.09:42:24 c5t03wz.log
L1
L2
L3
L4
L5
L6
L7
L8
L9
L10
L11
L12
L13
L14
U1
U2
U3
U4
U5
U6
U7
U8
U9
U10
U11
U12
U13
U14
-20 0
0
5
10
15
Baseband Frequency (MHz)
Baseband Frequency (MHz)

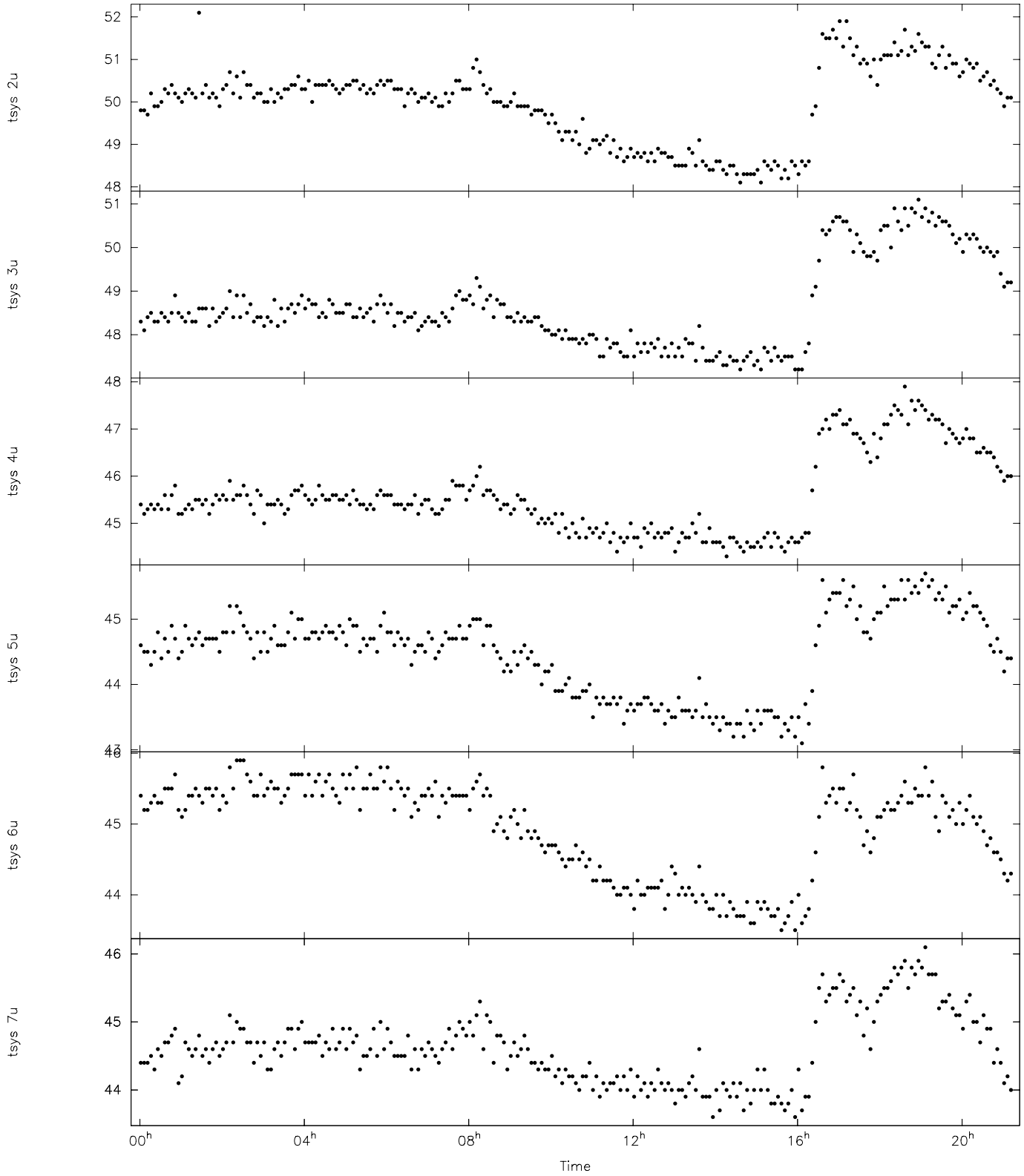

TIGOCONC Baseband Image Rejection (dB) -- 2005.214.20:04:57 c5t04atc.log
L1
L2
L3
L4
L5
L6
L7
L8
L9
L10
L11
L12
L13
L14
U1
U2
U3
U4
U5
U6
U7
U8
U9
U10
U11
U12
U13
U14
-20 0
0
5
10
15
Baseband Frequency (MHz)
Baseband Frequency (MHz)

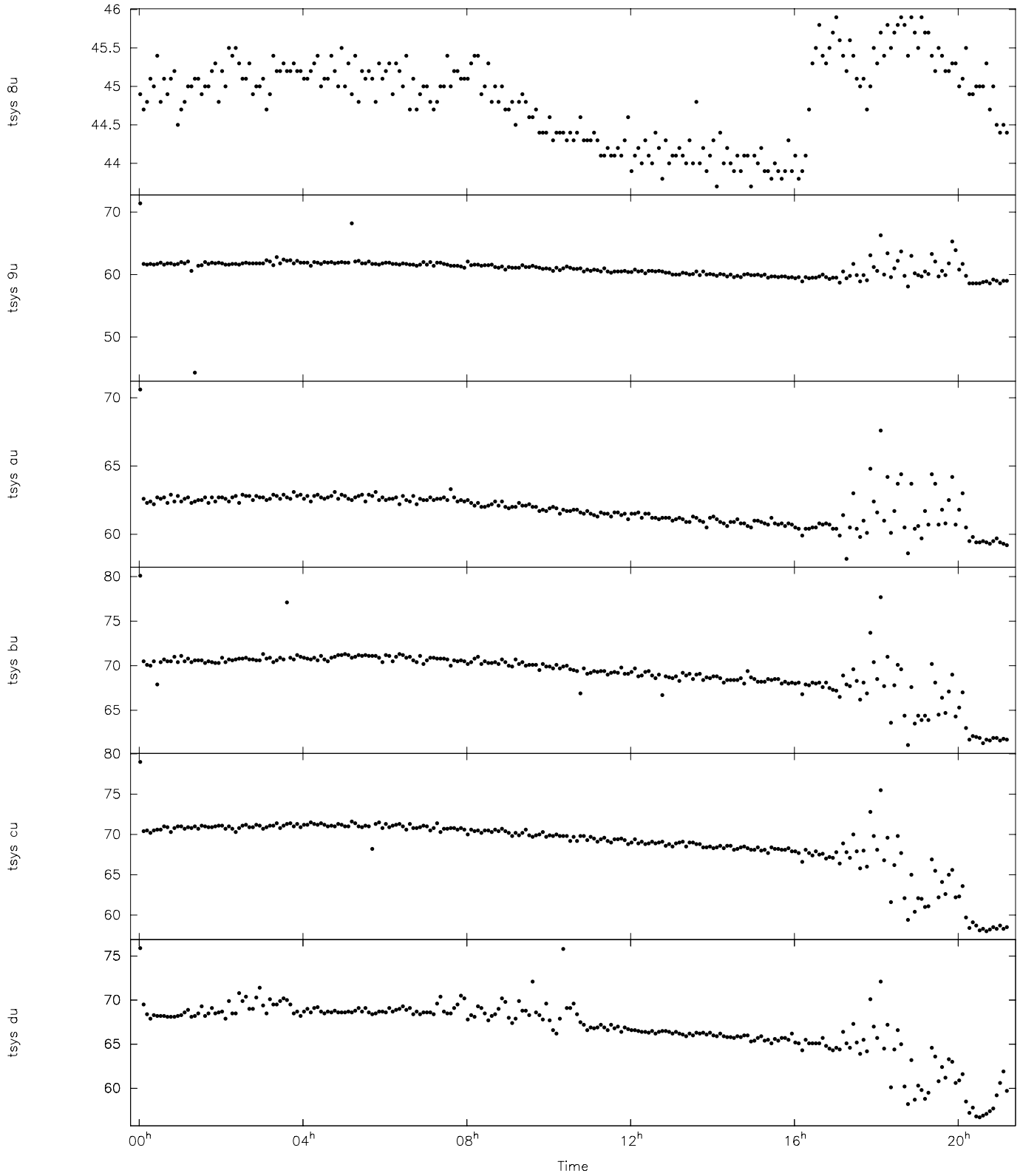

GILCREEK Sampler Statistics Deviation (percent) -- 2005.244.02:32:45 c5t05gc.log
L1
L2
L3
L4
L5
L6
L7
L8
L9
L10
L11
L12
L13
L14
U1
U2
U3
U4
U5
U6
U7
U8
U9
U10
U11
U12
U13
U14
10
0
10
−−
−
+
++
States
States

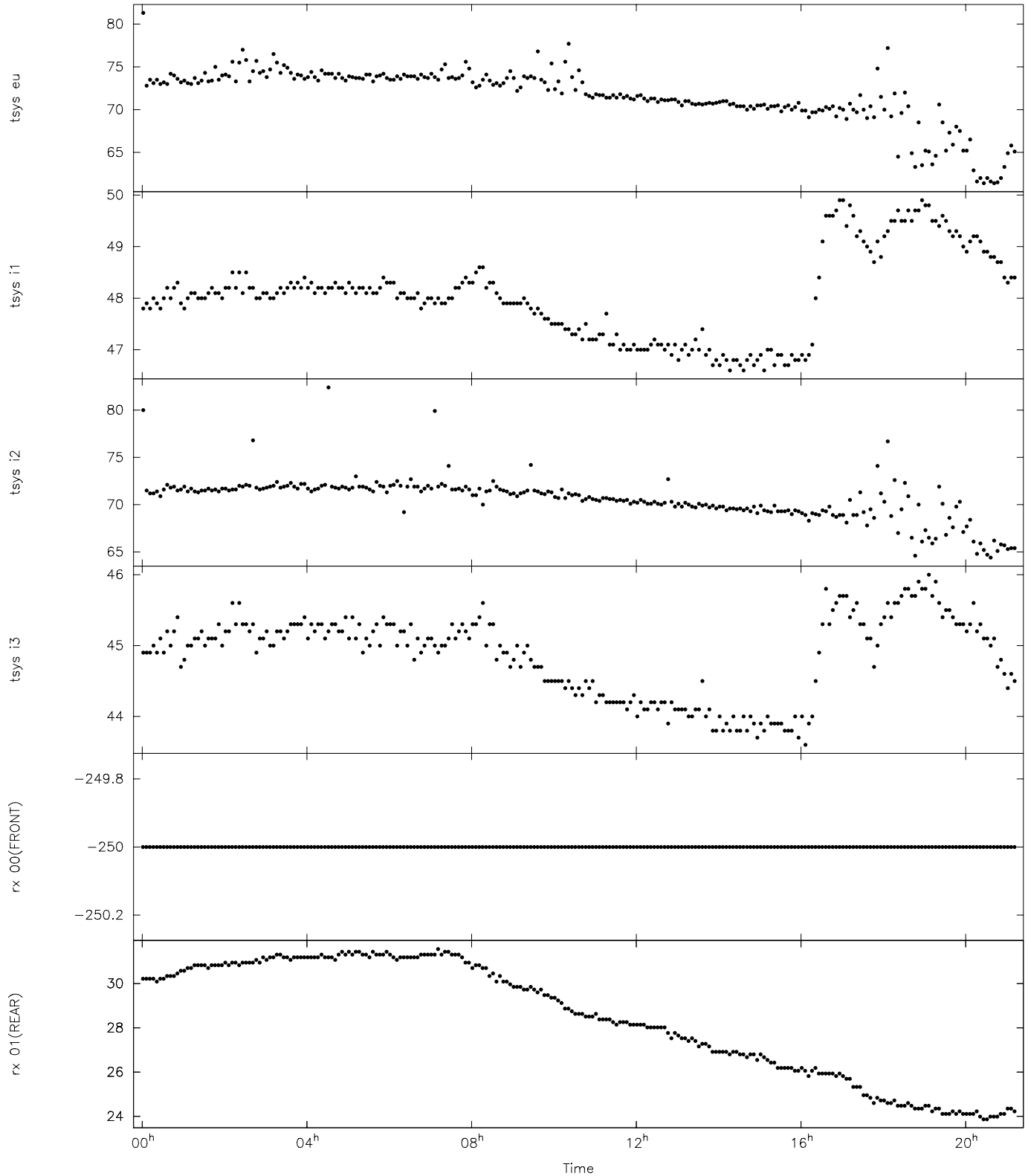

Temperature
20
15
Pressure
956.5
956
955.5
Humidity
100
90
80
70
Cable Delay (ps)
20
0
gps-fmout (us)
15.5
15.45
15.4
10000 Hz Amp U1
42
40
38
00h
04h
08h
12h
16h
20h
Time

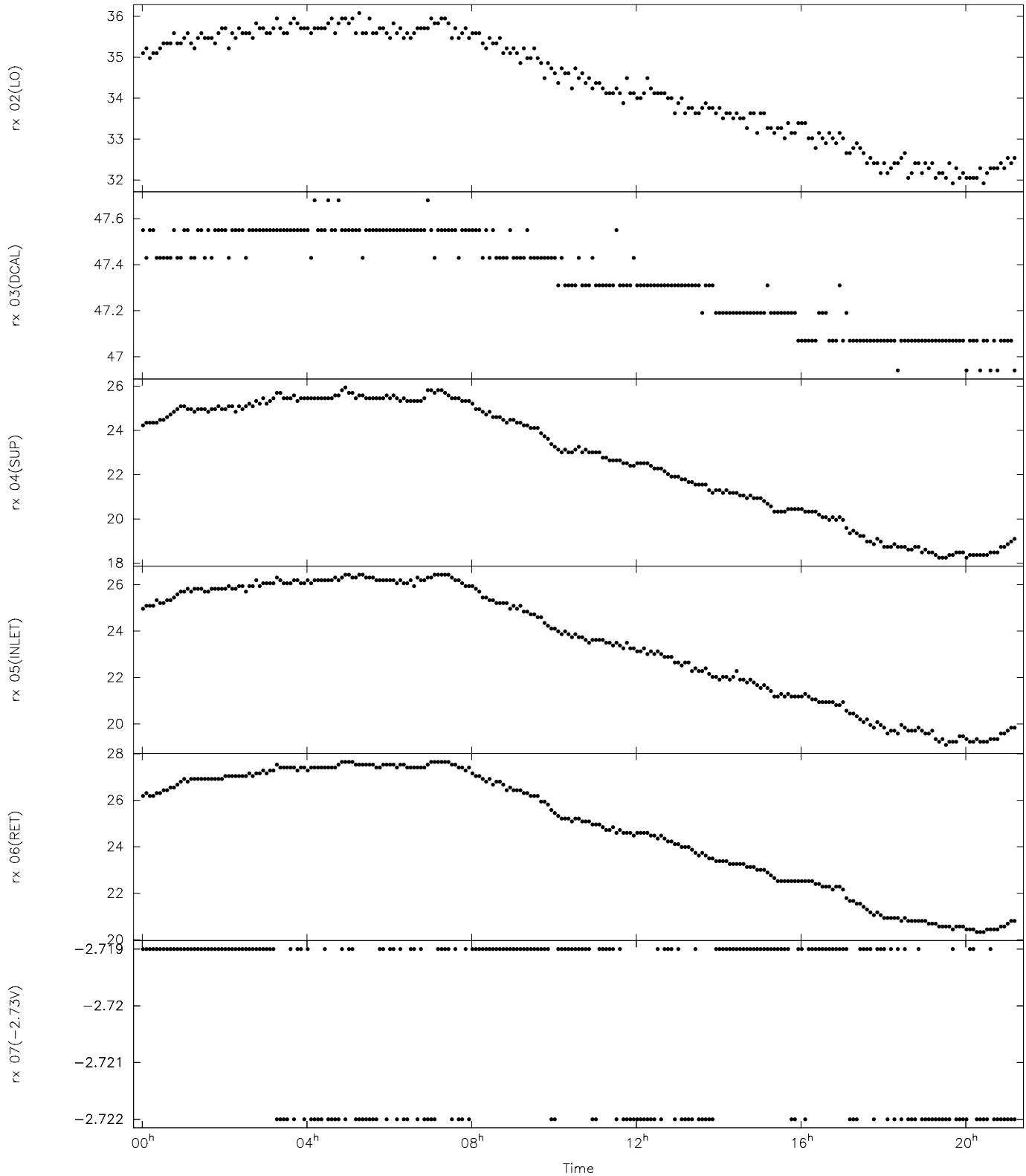

10000 Hz Phase U1
100
0
−100
10000 Hz Amp U4
40
30
20
10
0
10000 Hz Phase U4
100
0
−100
10000 Hz Amp U5
42
41
40
39
10000 Hz Phase U5
100
0
−100
10000 Hz Amp U8
44
43
42
41
40
00h
04h
08h
12h
16h
20h
Time

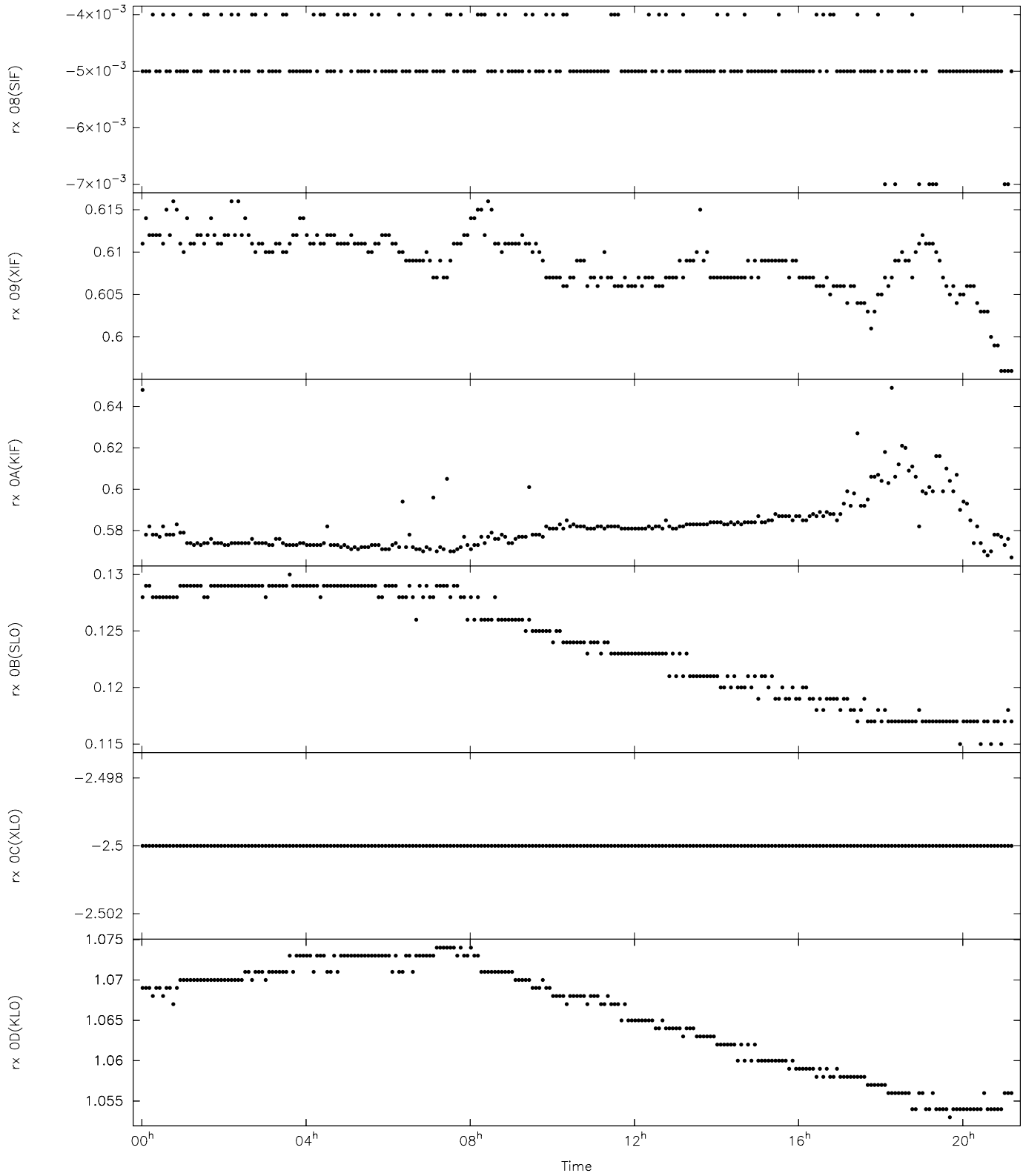

10000 Hz Phase U8
100
0
−100
10000 Hz Amp U9
45
40
35
10000 Hz Phase U9
0
−50
10000 Hz Amp U14
40
38
36
10000 Hz Phase U14
0
−50
−100
tsys 1u
53
52
51
50
00h
04h
08h
12h
16h
20h
Time

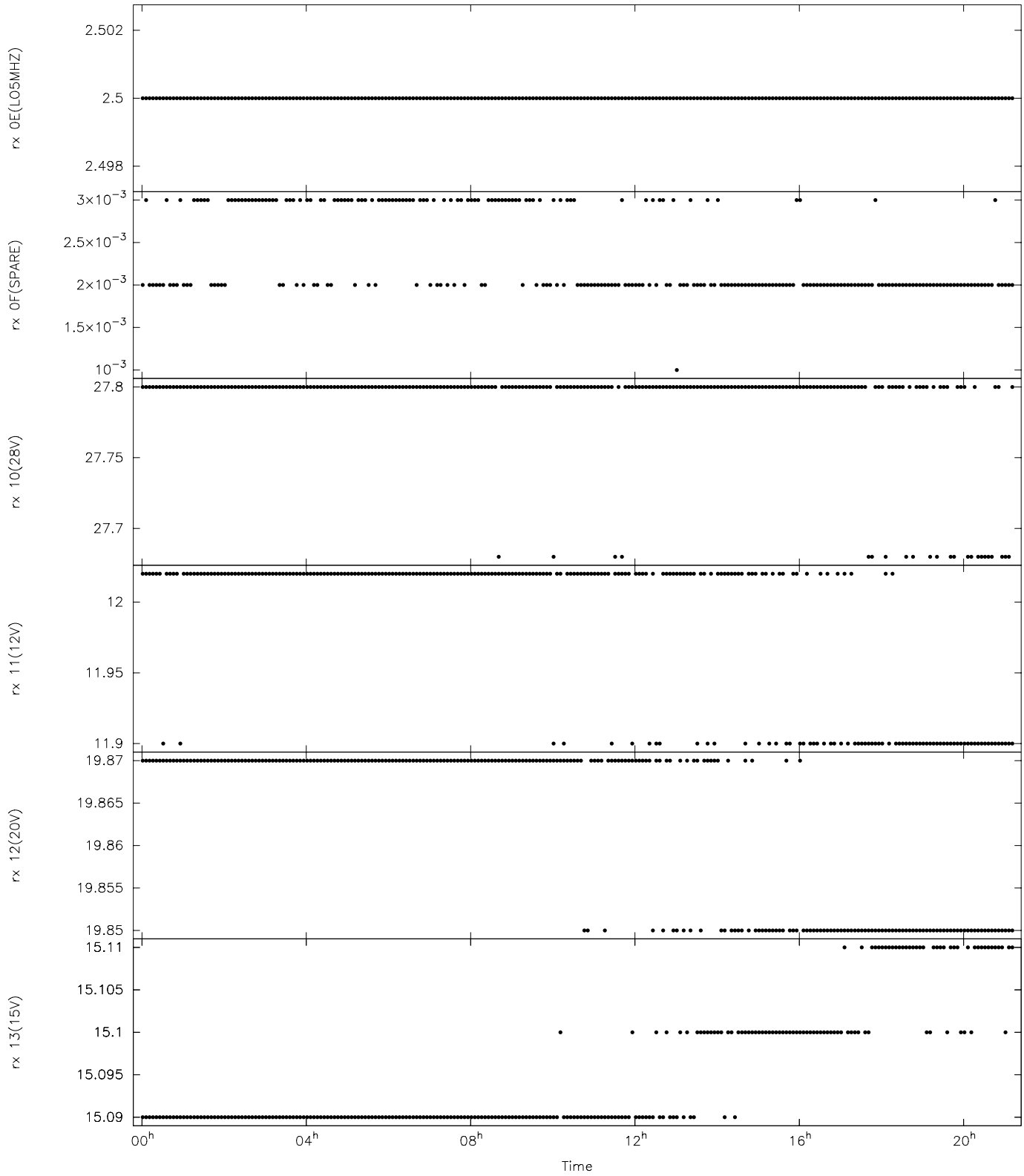

tsys 2u
tsys 3u
tsys 4u
tsys 5u
tsys 6u
tsys 7u
00h
04h
08h
12h
16h
20h
Time

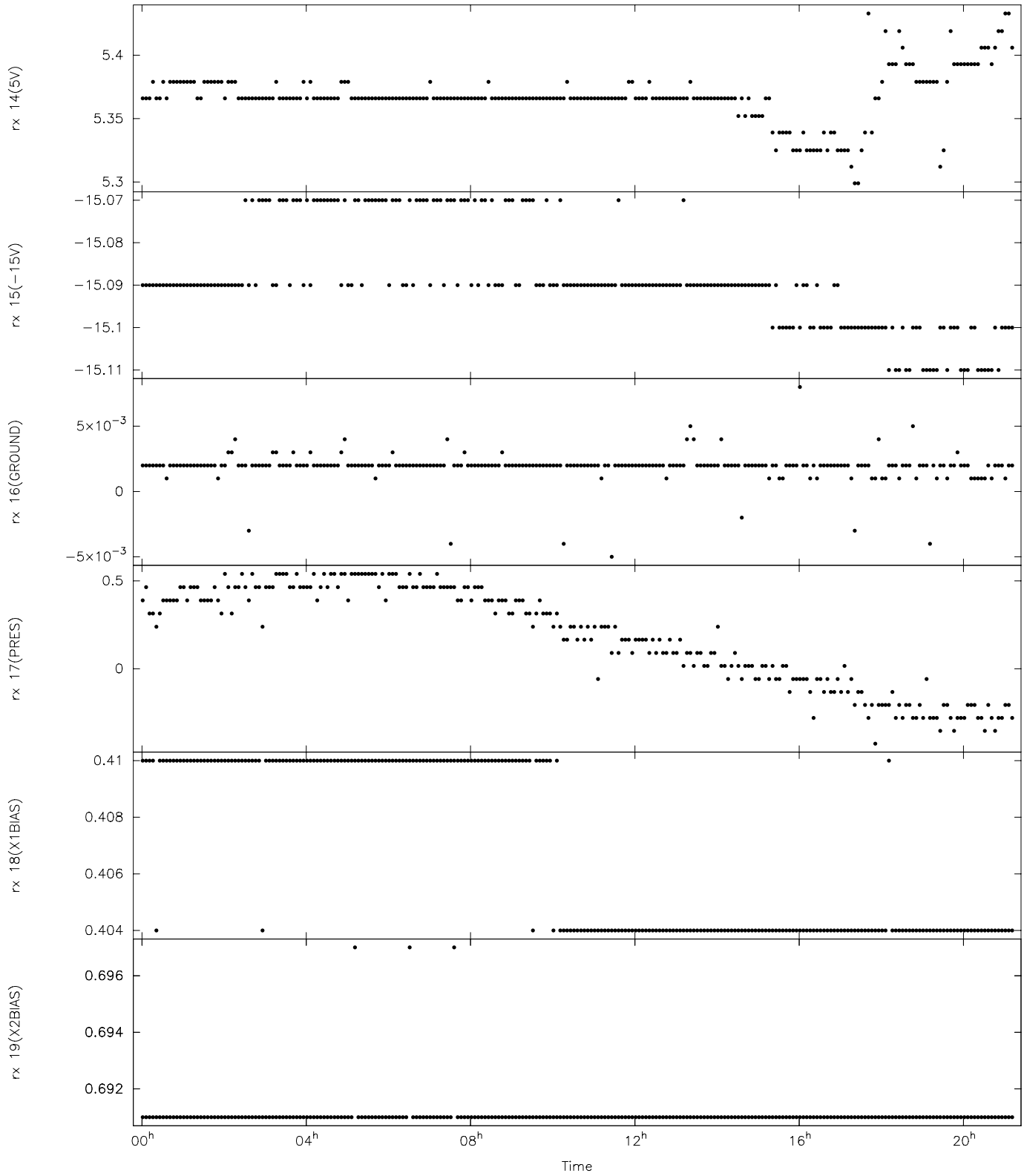

tsys 8u
46
45.5
45
44.5
44
tsys 9u
70
60
50
tsys au
70
65
60
tsys bu
80
75
70
65
80
tsys cu
75
70
65
60
tsys du
75
70
65
60
00h
04h
08h
12h
16h
20h
Time

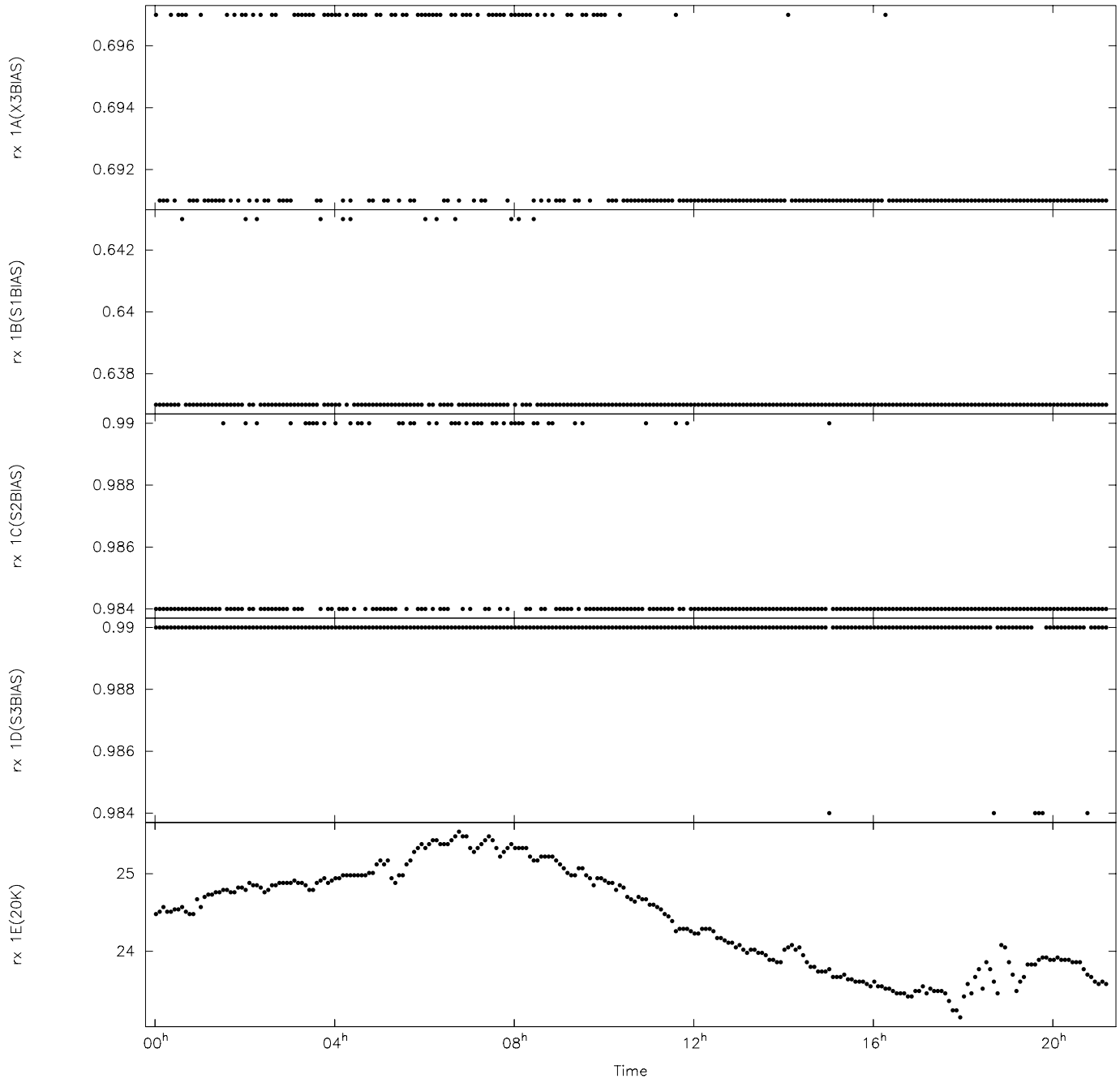

tsys eu
80
75
70
65
50
tsys i1
49
48
47
tsys i2
80
75
70
65
tsys i3
46
45
44
rx 00(FRONT)
−249.8
−250
−250.2
rx 01(REAR)
30
28
26
24
00h
04h
08h
12h
16h
20h
Time

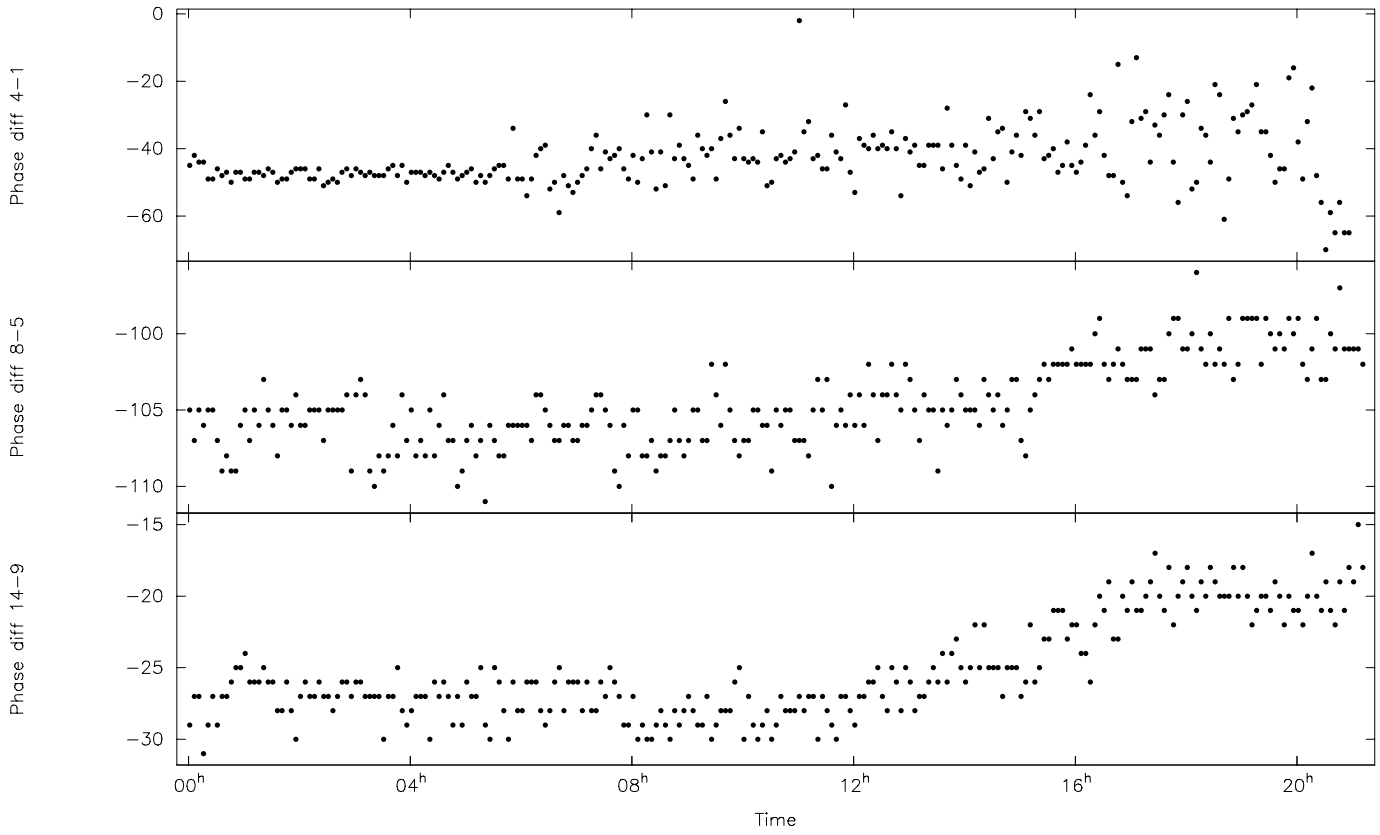

rx 02(LO)
36
35
34
33
32
rx 03(DCAL)
47.6
47.4
47.2
47
rx 04(SUP)
26
24
22
20
18
rx 05(INLET)
26
24
22
20
rx 06(RET)
28
26
24
22
20
rx 07(−2.73V)
−2.719
−2.72
−2.721
−2.722
00h
04h
08h
12h
16h
20h
Time

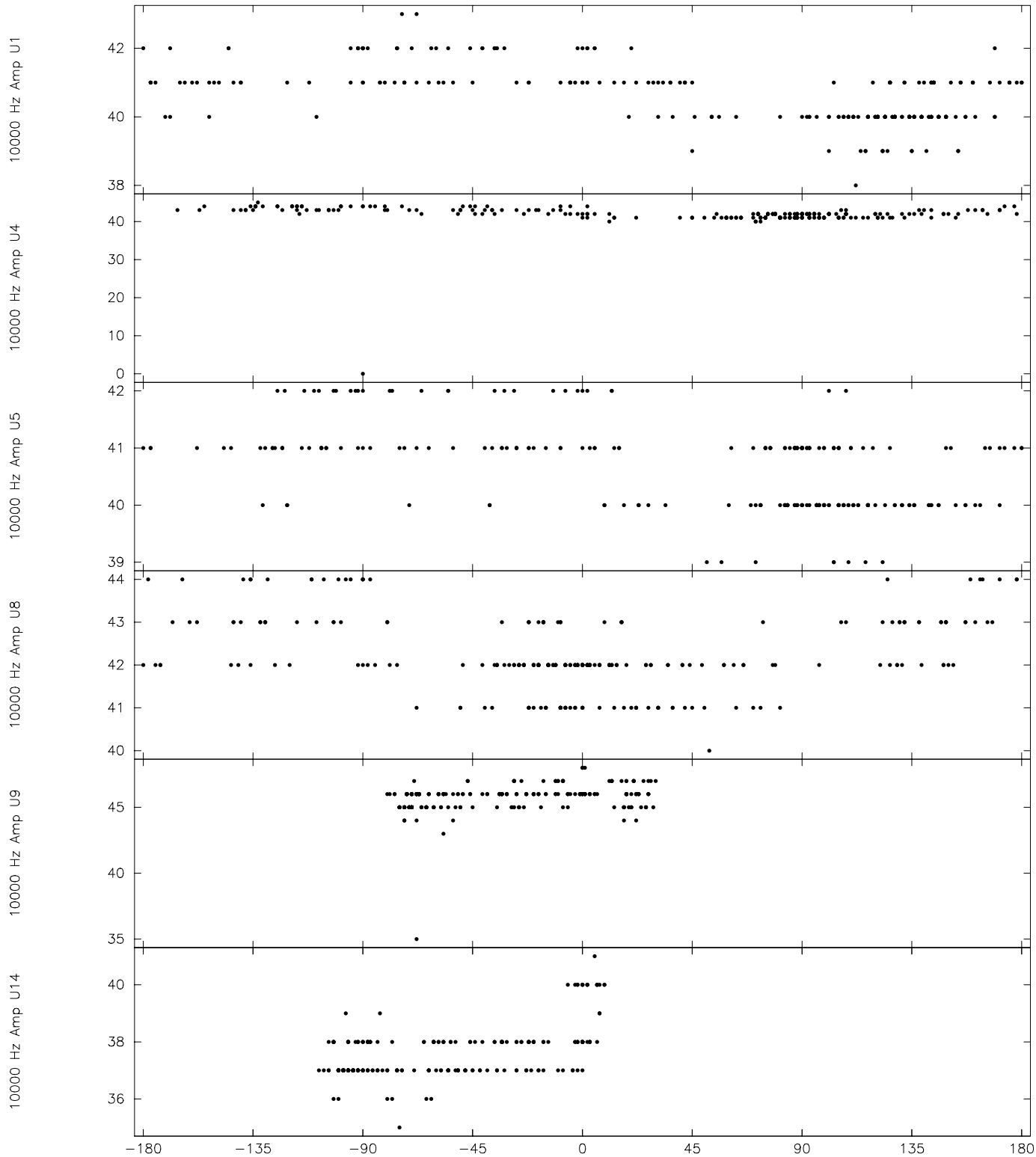

rx 08(SIF)
−4×10−3
−5×10−3
−6×10−3
−7×10−3
rx 09(XIF)
0.615
0.61
0.605
0.6
rx 0A(KIF)
0.64
0.62
0.6
0.58
rx 0B(SLO)
0.13
0.125
0.12
0.115
rx 0C(XLO)
−2.498
−2.5
−2.502
rx 0D(KLO)
1.075
1.07
1.065
1.06
1.055
00h
04h
08h
12h
16h
20h
Time


rx 0E(LO5MHZ)
2.502
2.5
2.498
rx 0F(SPARE)
3×10⁻³
2.5×10⁻³
2×10⁻³
1.5×10⁻³
10⁻³
rx 10(28V)
27.8
27.75
27.7
rx 11(12V)
12
11.95
11.9
rx 12(20V)
19.87
19.865
19.86
19.855
19.85
rx 13(15V)
15.11
15.105
15.1
15.095
15.09
00h
04h
08h
12h
16h
20h
Time


rx 14(5V)
5.4
5.35
5.3
rx 15(−15V)
−15.07
−15.08
−15.09
−15.1
−15.11
rx 16(GROUND)
5×10^−3
0
−5×10^−3
rx 17(PRES)
0.5
0
0.41
rx 18(X1BIAS)
0.408
0.406
0.404
rx 19(X2BIAS)
0.696
0.694
0.692
00h
04h
08h
12h
16h
20h
Time


rx 1A(X3BIAS)
0.696
0.694
0.692
rx 1B(S1BIAS)
0.642
0.64
0.638
rx 1C(S2BIAS)
0.99
0.988
0.986
0.984
rx 1D(S3BIAS)
0.99
0.988
0.986
0.984
rx 1E(20K)
25
24
00h
04h
08h
12h
16h
20h
Time


Phase diff 4−1
0
−20
−40
−60
Phase diff 8−5
−100
−105
−110
Phase diff 14−9
−15
−20
−25
−30
00h
04h
08h
12h
16h
20h
Time


10000 Hz Amp U1
42
40
38
10000 Hz Amp U4
40
30
20
10
0
10000 Hz Amp U5
42
41
40
39
10000 Hz Amp U8
44
43
42
41
40
10000 Hz Amp U9
45
40
35
10000 Hz Amp U14
40
38
36
−180
−135
−90
−45
0
45
90
135
180
Phase

chk1024 Examples

This is a listing of some sample chk1024 outputs for various logs. The command line used to generate each output is shown so that the options selected can be seen. All of the logs examined were generated before the decoder firmware version was recorded in the log. Consequently, all have the warning about the firmware. There are nine examples.

1. Successful check, except 4f, which was fixed later.

```
   chk1024  chk1024feb07a.log

Location: MEDICINA, Date: 2007.033.14:00:03, Log: chk1024feb07a.log
Setup `4f' tested 32 tracks, 0 were good (0%).
Setup `1g' tested 64 tracks, 64 were good (100%).
Setup `512' tested 64 tracks, 64 were good (100%).
Decoder firmware version number NOT detected.
```

2. Successful check, but checks were run more than once (more than a year apart) in this log. On the second run (first was discarded), duplicate DQA records occurred, but it was caused by the checks being used twice on the second run. It is not related to AUX data being missing.

```
   chk1024 chk1024_rd0507on_20050712.log

Location: Onsala  , Date: 2004.084.08:05:42, Log: chk1024_rd0507on_20050712.log
Location: Onsala  , Date: 2005.193.13:21:50, Log: chk1024_rd0507on_20050712.log
Setup 4f seen again, discarding previous results.
Setup 512 seen again, discarding previous results.
Setup 1g seen again, discarding previous results.
Setup `4f' tested 32 tracks, 32 were good (100%).
Setup `1g' tested 64 tracks, 64 were good (100%).
Setup `512' tested 64 tracks, 64 were good (100%).
Decoder firmware version number NOT detected.
Duplicate DQA records exist for some tracks, bad AUX checks?
```

3. Same log, but now with "-l" to show in which run different setups were used.

```
   chk1024 -l chk1024_rd0507on_20050712.log

Location: Onsala  , Date: 2004.084.08:05:42, Log: chk1024_rd0507on_20050712.log
Set-up 4f encountered
Set-up 512 encountered
Set-up 1g encountered
Location: Onsala  , Date: 2005.193.13:21:50, Log: chk1024_rd0507on_20050712.log
Set-up 4f encountered
Setup 4f seen again, discarding previous results.
Set-up 512 encountered
Setup 512 seen again, discarding previous results.
Set-up 1g encountered
Setup 1g seen again, discarding previous results.
Setup `4f' tested 32 tracks, 32 were good (100%).
Setup `1g' tested 64 tracks, 64 were good (100%).
Setup `512' tested 64 tracks, 64 were good (100%).
Decoder firmware version number NOT detected.
Duplicate DQA records exist for some tracks, bad AUX checks?
```

4. This one seems successful but is not because scan_check failed.

```
chk1024 chk1024e.log

Location: HartRAO , Date: 2007.050.12:57:27, Log: chk1024e.log
Setup `4f' tested 32 tracks, 32 were good (100%).
Setup `1g' tested 64 tracks, 64 were good (100%), ERROR: scan_check failed.
Setup `512' tested 64 tracks, 64 were good (100%).
Decoder firmware version number NOT detected.
```

5. Results appear bad due to old decoder firmware. This is the first example of "-s" option output. Note good tracks for "1g" is only 49 even though only three tracks were bad for "play" and none for "idle". This is because two were not "reported" (i.e., not seen in log) for "idle" and 10 for "play". There could be some overlap among these tracks, but in this case they add up to 15 (=64-49). Similar arithmetic works for other modes.

```
chk1024 -s chk1024-CMVA.log

Location: MATERA20, Date: 2007.038.10:37:28, Log: chk1024-CMVA.log
Setup Mode Total  Frames Errors  Aux   Set1odd  Set1even  Set2odd  Set2even
           Tracks ----------------------bad/reported-----------------------
  4f idle    32    0/32   0/32   0/32    0/16      0/16     0/ 0      0/ 0
  4f play    32    0/30   0/30   0/32    0/15      0/15     0/ 0      0/ 0
  1g idle    64    0/62   0/62   0/64    0/15      0/15     0/16      0/16
  1g play    64    0/54   3/54   2/64    2/12      1/12     0/15      0/15
 512 idle    64    0/46   0/46   0/64    0/11      0/11     0/12      0/12
 512 play    64    0/46   2/46   2/64    1/13      1/13     0/10      0/10
Frame check used a minimum of 300 (0x12c) frames/track.
Setup `4f' tested 32 tracks, 30 were good (93%).
Setup `1g' tested 64 tracks, 49 were good (76%).
Setup `512' tested 64 tracks, 30 were good (46%).
Decoder firmware version number NOT detected.
```

6. Old decoder firmware, same example as (5) above, but output improved somewhat with "-p" option. Only current firmware, 5.2 or later, should be used.

```
chk1024 -ps chk1024-CMVA.log

Location: MATERA20, Date: 2007.038.10:37:28, Log: chk1024-CMVA.log
Setup Mode Total  Frames Errors  Aux   Set1odd  Set1even  Set2odd  Set2even
           Tracks ----------------------bad/reported-----------------------
  4f idle    32    0/32   0/32   0/32    0/16      0/16     0/ 0      0/ 0
  4f play    32    0/32   0/32   0/32    0/16      0/16     0/ 0      0/ 0
  1g idle    64    0/64   0/64   0/64    0/16      0/16     0/16      0/16
  1g play    64    0/64   3/64   2/64    2/16      1/16     0/16      0/16
 512 idle    64    0/64   0/64   0/64    0/16      0/16     0/16      0/16
 512 play    64    0/64   2/64   2/64    1/16      1/16     0/16      0/16
Frame check used a minimum of 300 (0x12c) frames/track.
Setup `4f' tested 32 tracks, 32 were good (100%).
Setup `1g' tested 64 tracks, 61 were good (95%).
Setup `512' tested 64 tracks, 62 were good (96%).
Decoder firmware version number NOT detected.
```

7. Same station as examples (5) and (6) above, but now with correct firmware. There are still some small problems.

```
   chk1024 -s chk1024-CMVA-5_2.log

Location: MATERA20, Date: 2007.043.09:08:50, Log: chk1024-CMVA-5_2.log
Setup Mode Total  Frames Errors  Aux   Set1odd  Set1even  Set2odd  Set2even
           Tracks ----------------------bad/reported-----------------------
  4f idle    32    0/32   0/32   0/32    0/16      0/16     0/ 0      0/ 0
  4f play    32    0/32   0/32   0/32    0/16      0/16     0/ 0      0/ 0
  1g idle    64    0/64   0/64   0/64    0/16      0/16     0/16      0/16
  1g play    64    0/64   1/64   0/64    1/16      0/16     0/16      0/16
 512 idle    64    0/64   0/64   0/64    0/16      0/16     0/16      0/16
 512 play    64    0/64   1/64   0/64    1/16      0/16     0/16      0/16
Frame check used a minimum of 300 (0x12c) frames/track.
Setup `4f' tested 32 tracks, 32 were good (100%).
Setup `1g' tested 64 tracks, 63 were good (98%), ERROR: scan_check failed.
Setup `512' tested 64 tracks, 63 were good (98%).
Decoder firmware version number NOT detected.
```

8. Station with decoder communication problem (but firmware is okay) causing some AUX records to be missing, also bad AUX data generated for many tracks by formatter.

```
   chk1024 -s chk1024sv07100.log

Location: SVETLOE , Date: 2007.100.10:24:02, Log: chk1024sv07100.log
Setup Mode Total  Frames Errors  Aux   Set1odd  Set1even  Set2odd  Set2even
           Tracks ----------------------bad/reported-----------------------
  4f idle    32    0/32  16/32  14/18    9/ 9      5/ 9     0/ 0      0/ 0
  4f play    32    0/32   2/32  16/22    8/11      8/11     0/ 0      0/ 0
  1g idle    64    0/64  32/64  22/26    4/ 4      3/ 4     9/ 9      6/ 9
  1g play    64    0/64   0/64  28/38   11/14     11/14     3/ 5      3/ 5
 512 idle    64    0/64   3/64  36/46    7/ 8      7/ 8    11/15     11/15
 512 play    64    0/64   3/64  32/42   12/15     12/15     4/ 6      4/ 6
Frame check used a minimum of 300 (0x12c) frames/track.
Setup `4f' tested 32 tracks, 3 were good (9%).
Setup `1g' tested 64 tracks, 2 were good (3%).
Setup `512' tested 64 tracks, 6 were good (9%).
Decoder firmware version number NOT detected.
Duplicate DQA records exist for some tracks, bad AUX checks?
Some AUX records were missing.
```

9. Same log as example (8) above, but problems with AUX data results ignored with "-a" option.

```
   chk1024 -sa chk1024sv07100.log

Location: SVETLOE , Date: 2007.100.10:24:02, Log: chk1024sv07100.log
Setup Mode Total  Frames Errors  Aux   Set1odd  Set1even  Set2odd  Set2even
           Tracks ----------------------bad/reported-----------------------
  4f idle    32    0/32  16/32  14/18   16/16      0/16     0/ 0      0/ 0
  4f play    32    0/32   2/32  16/22    2/16      0/16     0/ 0      0/ 0
  1g idle    64    0/64  32/64  22/26   16/16      0/16    16/16      0/16
  1g play    64    0/64   0/64  28/38    0/16      0/16     0/16      0/16
 512 idle    64    0/64   3/64  36/46    2/16      0/16     1/16      0/16
 512 play    64    0/64   3/64  32/42    0/16      1/16     0/16      2/16
Frame check used a minimum of 300 (0x12c) frames/track.
Setup `4f' tested 32 tracks, 16 were good (50%).
Setup `1g' tested 64 tracks, 32 were good (50%).
Setup `512' tested 64 tracks, 58 were good (90%).
Decoder firmware version number NOT detected.
Duplicate DQA records exist for some tracks, bad AUX checks?
Some AUX records were missing.
AUX data ignored (option -a) for 'Set' summaries and overall statistics.
```